Pre-Estimate Burst Scheduling (PEBS): An efficient architecture with low realization complexity for burst scheduling disciplines

Sascha Junghans
University of Stuttgart
Institute of Communication Networks and Computer Engineering (IKR)
Stuttgart, Germany
Email: junghans@ikr.uni-stuttgart.de
Telephone: +49 711/685-7974
Fax: +49 711/685-7983

***Abstract*— Optical Burst Switching (OBS) [1] is a promising candidate for a more dynamic optical network layer. The scheduling of bursts in OBS core nodes is an important task for achieving good network performance. A lot of scheduling algorithms were published, but so far only few work was done on analyzing the realization complexity and on building prototypes for scheduling modules. In this paper we present for the first time the new scheduling scheme Pre-Estimate Burst Scheduling (PEBS) which reduces the realization complexity of scheduling modules by pre-calculating parts of the scheduling decisions.**

I. The task of burst scheduling

Optical Burst Switching is characterized by an out-of-band signalling scheme. The header information is sent as burst header packet (BHP) on a explicit control channel while the burst data itself is sent on several data channels. As the burst is not delayed in the node, while the BHP must be processed before the corresponding data arrives, the BHP is sent out a time period, called processing time compensating offset (PC-Offset), in advance to the burst. Different QoS-classes can be supported by the introduction of additional QoS-offsets for high priority bursts.

Fig. 1 shows the architecture of an OBS core node. The node can be separated in two parts: The optical domain with the optical cross connect (OXC) in the lower part which is not considered in detail and the optical switch controller part which is explained here. The control channel is forwarded to the input port controller (IPC) of the control part of the node. Here, it is converted into the electrical domain and a timestamp with the actual local system time is assigned to the BHP.

BHP processing in the core nodes is done in several steps: 1) the BHP (and its corresponding burst data) is assigned to an output port and forwarded to this port by the electrical cross connect (EXC), 2) in the output port controller (OPC), the bursts must be scheduled to a wavelength in a scheduling module. 3) The scheduling decision must then be communicated to the OXC calendar and a new BHP must be generated to inform the following nodes on the path. The burst scheduling module must assign the burst to a wavelength and keep track of all scheduled bursts and store the status of the wavelengths until the burst passed the node.

The scheduling process itself contains three tasks with a high degree of interaction:

- All wavelengths which would be able to transmit the new burst must be identified.
- From the identified wavelengths one must be selected and the new burst is assigned to this wavelength
- A bookkeeping mechanism must keep track of all burst assignments for the identification of free wavelengths for following burst scheduling requests.

Scheduling algorithms for OBS can be classified in so-called *horizon-based* algorithms, which only store the latest point in time, for which each wavelength is used and in algorithms with *void-filling*, where bursts can be assigned between two scheduled bursts if the void is large enough and the burst arrives in the right moment. As void-filling algorithms reserve the wavelength for exact the time period when the burst passes the node, they are often called *Just-Enough-Time (JET)* algorithms [2].

For the wavelength selection process several algorithms are introduced in literature: If the first free wavelength is used the algorithm is called *First Fit (FF)*. Other algorithms rate the assignment to each wavelength by predefined criteria and then select the wavelength with the highest or lowest rating. The scheduling algorithm *latest available unscheduled channel with void filling (LAUC-VF)* [3] tries to minimize the resulting gap between the new burst and the burst scheduled on this wavelength passing before the new one.

II. Realization of scheduling modules

Up to now, only few work on realization of scheduling modules is published. A software-based scheduling system was introduced in [4], which allows scheduling of bursts with burst lengths in the range of milliseconds. As we showed in [5] and [6], software based solutions are not suitable for network scenarios with smaller mean burst lengths. For these scenarios hardware based scheduling systems are necessary.

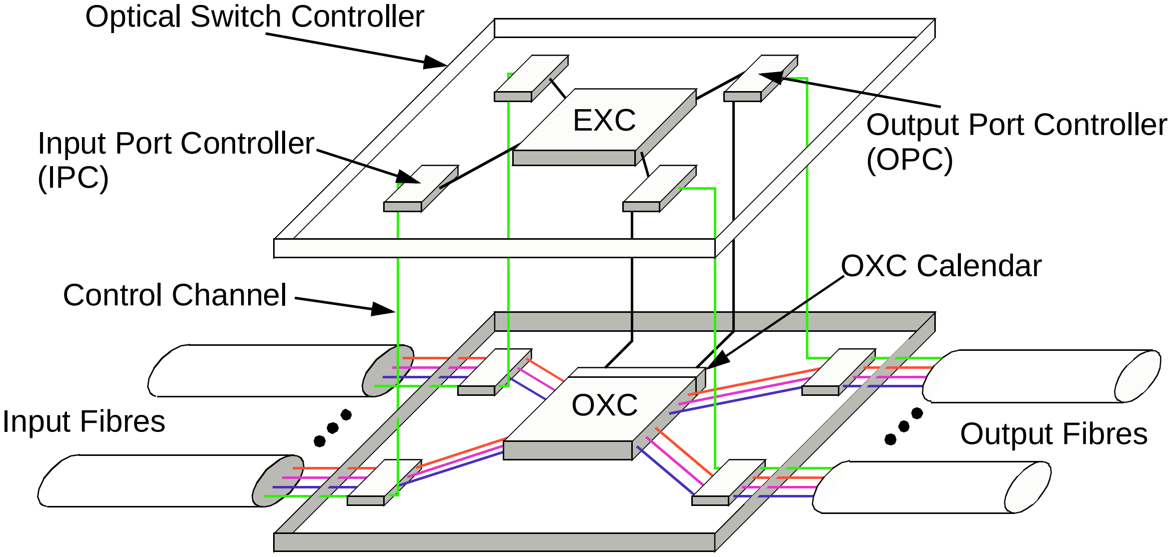
Optical Switch Controller
Input Port Controller (IPC)
EXC
Output Port Controller (OPC)
Control Channel
OXC Calendar
OXC
Input Fibres
Output Fibres

Fig. 1. Architecture of an OBS core node

For *horizon*-based scheduling we presented a hardware-based module in [7]. In [5], we showed, that the void-filling scheduling mechanism *First Fit* can be realized fast enough with todays Field Programmable Gate Array (FPGA) technology but needs a lot of resources. Especially algorithms which allow void filling with variable offset lengths need complex comparison or sorting operations for the identification of suitable wavelengths. This resource demand depends on several parameters.

With an increasing number of wavelengths the performance of the scheduling module is influenced: More wavelengths can transport more bursts which leads to a higher load of the scheduling module. Additional, the higher number of wavelengths leads to a higher complexity of the module, because more wavelengths must be considered in the identification, selection and bookkeeping process for each new burst.

The more bursts can be scheduled in advance, the more voids appear in each wavelength. For a new burst the transmission time of all bursts must be compared to identify the suitable voids. If the scheduling module processes them in sequential order, more bursts lead to more system cycles for the processing. If the comparison is performed in parallel the complexity raises because of the higher number of comparators and storage elements.

Further investigations showed, that the realization of more complex schemes like *LAUC-VF* are not realizable for larger number of wavelengths with the timing constraints identified in [5]. Besides the complex identification process needed for *First Fit*, *LAUC-VF* needs a large comparison stage for the selection process. This leads to a very high degree of meshed logic, which reaches limitations of on-chip routing resources and leads to slow system clocks.

III. Offset length distributions in OBS networks

Publications dealing with performance aspects of OBS scheduling assume very different traffic patterns for their performance evaluation. Some general rules for burst traffic parameters can be derived from these studies. From our experience with analytical and simulative studies for OBS [8], [9], [10], [11] we identified a realistic scenario with the following parameters:

- The assembly of a burst at the edge nodes takes more time than the transmission of the burst itself. Assuming that the link rates of the core network are about 10 times higher than the rates in the connected access networks, the assembly time is more than ten times larger than the transmission time of the burst. With an upper bound of a maximum assembly delay in the area of ten milliseconds, this lead to a mean burst length smaller than one millisecond.
- In [9] was shown, that for a good separation of Quality of Service-Classes in Offset-based QoS-Scenarios, the additional QoS-offset should be at least 2 - 5 times larger than the mean burst length. When a burst is ready for transmission, the BHP must be sent out and with the delay of the offset the burst is transmitted afterwards. As these QoS-offsets lead to additional delays of the bursts they should be kept as small as possible. This is another argument for small mean burst lengths which in turn lead to small QoS-offsets.
- PC-offsets vary on the way of the BHP through the OBS network. With large offset variations within each QoS class an undesired effect occurs: Similar to the separation of QoS classes by different offsets, a separation of bursts in one QoS class occurs with varying offsets. Especially bursts with small offsets which means that they have already traversed a large distance in the network and so

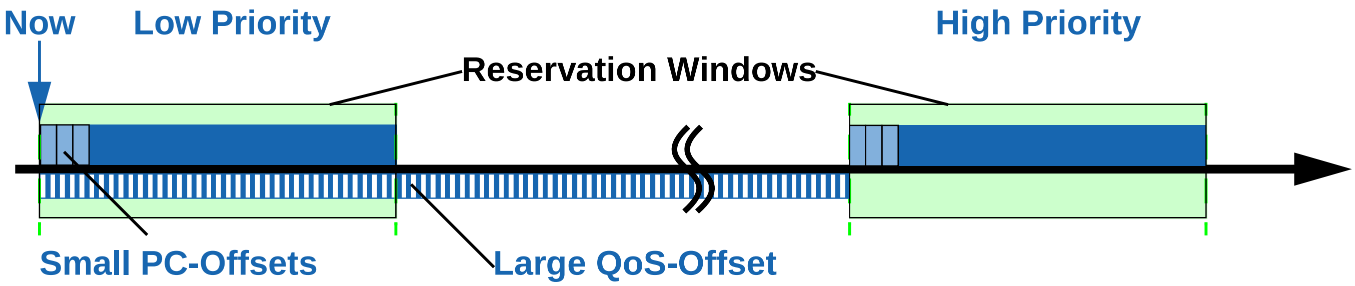

Fig. 2. Offset distributions and reservation windows

have already used many network resources are discarded with a higher probability than bursts with larger offsets and a shorter passed distance. This behaviour is not desirable. For minimizing this effect the PC-offsets should be kept as short as possible, 1% to 10% of the mean burst length would be good values.

- Two service classes are enough in OBS networks

The determination of a mean burst length value in the area of several 100 microseconds seams to be reasonable: The QoS-offset is then around one to several milliseconds and the PC-offset around one to several microseconds.

The discussion about offset lengths, above, lead to a significant observation. The offset length distribution is not a continuous one but has two distinct ranges: For low priority bursts the offsets lengths vary between $1{*}$*PC-offset* and $N_{max}{*}$*PC-offset* with N_{max} as the maximum number of hops in the OBS network. For high priority bursts the burst length distribution varies between *QoS-offset* + *PC-offset* and *QoS-offset* $+N_{max}{*}$ *PC-offset*. Together with the definition of a maximum burst length two distinct reservation windows are defined (Fig. 2). Only BHPs which announce bursts for these two windows should occur in the network and arrive at the *Optical Switch Controller*. This leads to a major advantage for the realization of scheduling modules. The scheduling module needs not to compare all reserved bursts with the data in the new BHP but only these who are located in one of the reservation windows.

For reaching a lower realization complexity a further step can be taken: The start and end times of a new burst are not compared with all reserved bursts in the reservation windows but only with the edges of the windows. When the suitable reservation window is identified, it is only checked, which wavelengths are not occupied in the suitable window. This occupation state needs not to be calculated in that moment but can be pre-calculated in advance: This mechanism *Pre-Estimated Burst Scheduling* (PEBS) has a much lower realization complexity than other scheduling modules for void-filling algorithms.

Pre-Estimate Burst Scheduling can be combined with existing wavelength selection schemes: For *First Fit*, a free wavelength can be identified for the two windows in advance, so that with the arrival of a BHP only the suitable window must be identified. For *LAUC-VF*, the wavelength which leaves the smallest void for a burst with minimal offset length can be pre-calculated as well.

IV. Performance of the PEBS module

For the realization of prototypical network nodes or parts of them, our institute designed and build up the so-called *Universal Hardware Platform (UHP)*. The platform allows the flexible integration of *Field Programmable Gate Arrays (FPGA)*, different types of memory and network interfaces like optical Gigabit-Ethernet. Based on this platform, we designed a testbed for OBS node controllers [7]. This setup allows the full realization of scheduling modules and the test with realistic traffic patterns without the complex operation of optical components. Fig. 3 shows a block diagram of the testbed.

The optical switch controller is realized on an *Altera-APEX20KC*-FPGA. BHPs are sent to it via optical Gigabit Ethernet. The *input port controller (IPC)* terminates the control channel, assigns a time stamp to arriving BHPs, identifies the target *output port controller (OPC)* and forwards the BHP to it via the electrical cross connect. Actually, only one IPC and one OPC are used, but all interfaces can cover more ports for future scenarios. Measurement components are integrated in the system and allow a detailed understanding of the impact of each component on the system performance.

For the generation of BHPs a traffic generator was realized. A PC calculates burst lengths and interarrival times with software generators from simulation library of the institute [12]. The parameters are sent in Ethernet packets to an FPGA. The FPGA extracts the parameters and generates the BHPs with the received parameters and sends the bursts with the exact interarrival times.

For the analysis of the scheduling results, a traffic analyzer was implemented. It receives all BHPs and forwards them to another PC for the analysis.

We realized the PEBS module for the support of 8 wavelengths and compared it with our realization of a *First Fit* module introduced in [5] regarding resource requirements, operation speed and network performance.

The *First Fit* module needs about 4 times more logic elements as the PEBS module (13.000 elements vs. 4500 elements). Main reason for this large difference is the fact, that the JET module must store the information of all bursts

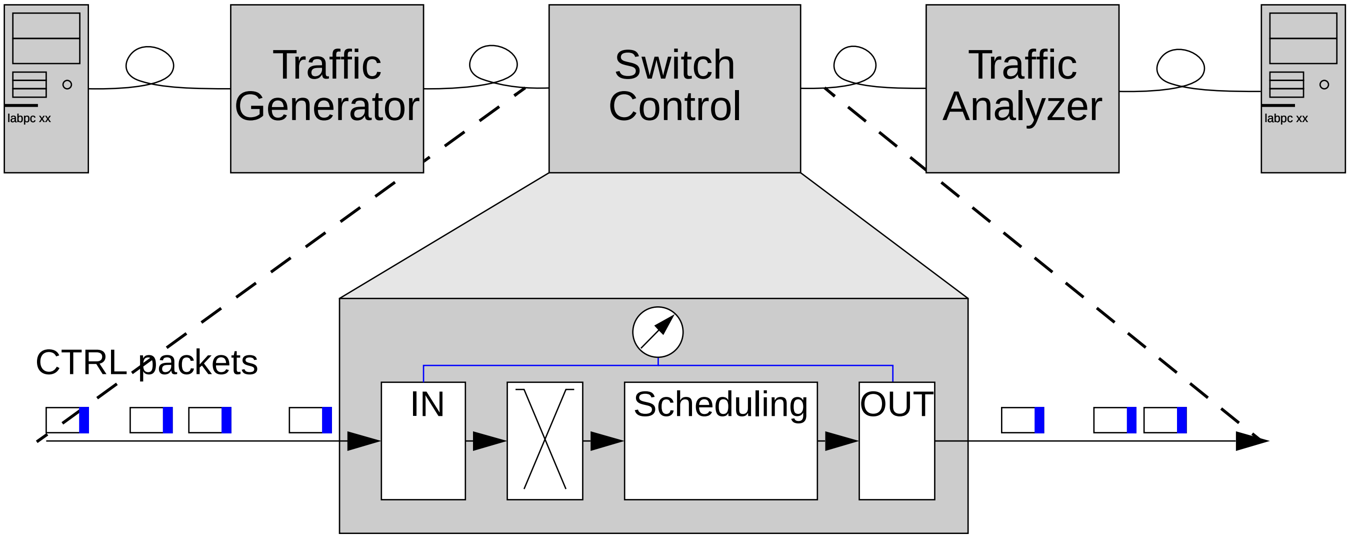

Fig. 3. Testbed for burst scheduling modules

in logic elements because the data of all reserved bursts is accessed in parallel. As PEBS does not consider bursts outside the reservation windows, the data of these bursts can be stored in memory blocks on the FPGA. These memory blocks must be added to the list of needed resources but are only 3% of the available FPGA memory resources.

Besides the storage elements, the *First Fit* module must compare all data of reserved bursts with the data of the new burst. Therefore a large amount of comparators must be placed on the FPGA which need a lot of timing critical resources. The results of the comparators must be combined for the wavelength assignment process. This leads to long combinatorial paths which limit the maximum operating frequency of the scheduling module. PEBS needs comparators as well but as the window edges move deterministic, the comparisons can be pre-calculated which allows to set up a small pipeline structure for the reservation process. This allows a higher system frequency. The PEBS module can operate with a system clock which is more than two times faster than the *First Fit* module (45 MHz vs. 21 MHz).

As this approach pre-calculates the target wavelength, the scheduling decision is not exact and leads to a reduced network performance compared to the original algorithms. The BHPs arrive with different PC-offsets at the module. As the module always assumes the PC-offset value zero, bursts which arrive with a larger offset could be rejected even if a wavelength would be free at the exact burst start time. In a similar way the PEBS module does not check the size of a void against the true end of the burst but against the worst-case end of the burst. Fig. 4 shows this behaviour: Wavelength WL n+1 would be able to transport the burst but due to the larger reservation window at the beginning of the burst the burst can not be accepted on this wavelength. Similar, WL n could carry the burst as well but the larger end time of the reservation window leads to a rejection of the burst, too. However, the wavelength status is stored with the exact times, i.e. the imprecision does not propagate and only the requested resources are reserved.

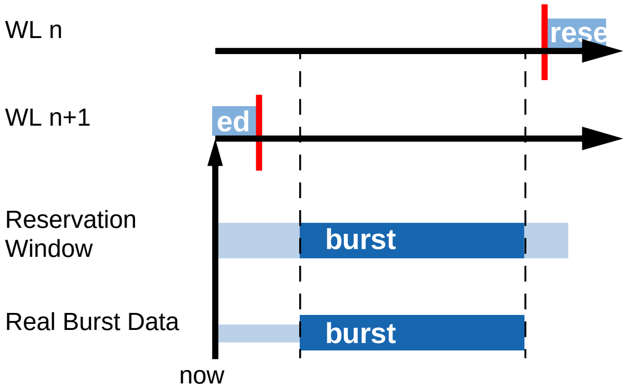

Fig. 4. Impact of reservation windows on burst reservation

The impact of the pre-calculated comparisons on the network performance can be seen in Fig. 5. We derived the results by measurements with the following parameters: maximum burst length: $150\mu s$, mean burst length: $100\mu s$, PC-offset: $5\mu s$, QoS-offset: $450\mu s$, maximum hops: 5 and 30% high priority traffic. The error bars show the standard deviation of 10 measurements per load. Burst lengths and number of hops are uniform distributed for both service classes. We show the burst loss probability for different network loads. As expected, the performance of PEBS is reduced with comparison to the *First Fit* module but still has a good network performance. The performance reduction is acceptable when we consider the realization complexity of the modules: For 8 wavelengths, both modules perform fast enough and can be synthesized on todays FPGA technology. As we showed in [6], the *First Fit* module reaches limitations in resources requirements and desired system frequencies at 64 wavelengths. For the PEBS module, the resource needs are lower and system clock is faster so that a higher number of wavelengths can be supported.

V. Conclusion

In this paper we explained the complexity issue of the realization of scheduling modules for OBS. The scenario for OBS networks we identified by extensive analytical and simulative performance evaluations allows the pre-calculation of comparisons needed for the wavelength identification process. This pre-calculation greatly reduces the realization complexity for scheduling modules. Here, we presented the first time our OBS scheduling scheme *Pre-Estimate Burst Scheduling (PEBS)* which needs less logic resources by higher system clock frequencies on the cost of a slightly reduced performance compared to *First Fit*. The burst scheduling module was realized on our FPGA based testbed and evaluated concerning realization complexity and network performance.

As further steps we will investigate PEBS in more detail concerning different reservation window sizes and scalability issues. The presented OBS network scenario allows additional scheduling concepts. We would like to present these concepts and to study their network performance and resource requirements in future.

References

[1] C. Qiao and M. Yoo, “Optical burst switching (OBS)—a new paradigm for an optical Internet,” *Journal of High Speed Networks*, vol. 8, no. 1, pp. 69–84, January 1999.

[2] M. Yoo and C. Qiao, “Just-enough-time (JET): A high speed protocol for bursty traffic in optical networks,” in *Digest of the IEEE/LEOS Summer Topical Meetings*, Montreal, Que. , Canada, August 1997, pp. 26–27.

[3] Y. Xiong, M. Vandenhoute, and H. Cankaya, “Design and analysis of optical burst-switched networks,” in *Proceedings of SPIE Conference on All Optical Networking*, vol. 3843, September 1999, pp. 112 – 119.

[4] J. Xu, C. Qiao, J. Li, and G. Xu, “Channel scheduling algorithms in optical burst switching networks,” in *IEEE Infocom*, San Francisco, April 2003.

[5] S. Junghans and C. M. Gauger, “Resource reservation in optical burst switching: Architectures and realizations for reservation modules,” in *OptiComm 2003*, Dallas, October 2003.

[6] C. M. Gauger and S. Junghans, “Architectures for resource reservation modules for optical burst switching core nodes,” in *4th ITG Symposium on Photonic Networks*, Leipzig/Germany, May 2003.

[7] S. Junghans, “A testbed for control systems of optical burst switching core nodes,” in *Proceedings of the Third International Workshop on Optical Burst Switching (WOBS)*, San Jose/CA, October 2004.

[8] K. Dolzer, C. M. Gauger, J. Späth, and S. Bodamer, “Evaluation of reservation mechanisms for optical burst switching,” *AEÜ International Journal of Electronics and Communications*, vol. 55, no. 1, pp. 18–26, January 2001.

[9] K. Dolzer and C. M. Gauger, “On burst assembly in optical burst switching networks—a performance evaluation of just-enough-time,” in *Proceedings of the 17th International Teletraffic Congress (ITC 17)*, Salvador, Brazil, December 2001, pp. 149–160.

[10] C. M. Gauger, K. Dolzer, , J. Späth, and S. Bodamer, “Service differentiation in optical burst switching networks,” in *2nd ITG Symposium on Photonic Networks*, January 2001, pp. 124 – 132.

[11] C. M. Gauger, “Performance of converter pools for contention resolution in optical burst switching.” in *Proceedings of the SPIE Optical Networking and Communications Conference (OptiComm)*, Boston, October 2002.

[12] “The IKR simulation library,” Institute of Communication Networks and Computer Engineering, University of Stuttgart, www.ikr.uni-stuttgart.de/IKRSimLib.

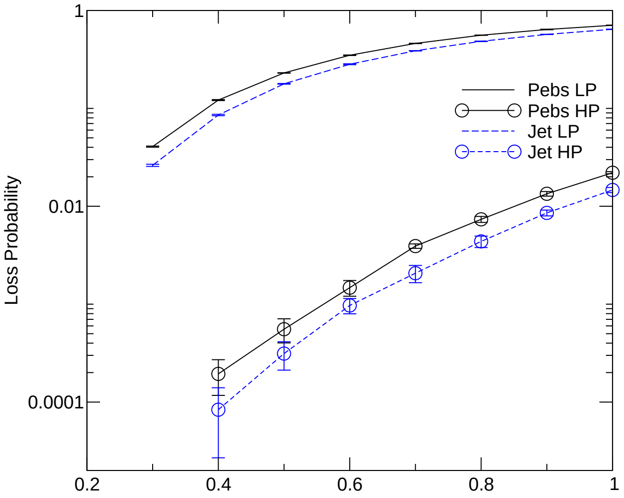

Fig. 5. Network performance of JET and PEBS