Optical Communications

Letter

Reservations inside clusters of all-optical core nodes achieve distributed burst aggregation and switching improving efficiency and loss

A. Stavdas¹, K. Kanonakis², G. Koukouvakis², H. C. Leligou²*, Th. Orphanoudakis¹ and J. D. Angelopoulos²

¹Department of Telecommunications Science and Technology, University of Peloponessse, Tripolis, Greece
²National Technical University of Athens, School of Electrical and Computer Engineering, Telecommunications Systems Laboratory, 9 Heroon Polytechniou Street, Zographou, 15773, Athens, Greece

SUMMARY

Clustering neighbouring nodes of an all-optical core network into medium-sized rings controlled by a master node allows two-way reservation-based control to aggregate bursts destined for other clusters in a lossless way with tolerable delay. The densely packed bursts can then be sent over static or switched light paths and be received in the destination cluster ring without the delay of end-to-end reservations. This hybrid approach where two-way reservations are geographically limited inside clusters only avoids both the handicap of heavy loss of one-way Optical Burst Switching (OBS) and the intolerable delay and multiplexing gain of end-to-end reservations. In addition, the number of multi-port space-switching all-optical nodes is reduced delegating part of this function to the distributed control of the laser sources of the ring under the guidance of the medium access protocol. Copyright © 2007 John Wiley & Sons, Ltd.

1. INTRODUCTION AND MOTIVATION

Wavelength routed networks, though more flexible than static links, lack the dynamics to handle modern bursty traffic efficiently. Optical Burst Switching (OBS) on the other hand suffers from excessive losses [1], because of collisions whenever two or more bursts contend for the same output port. Despite the presence of multiple WDM channels and possibly limited optical buffering, losses grow with every node crossed (the probability of success is the joint probability of success in all nodes). The need to retransmit lost bursts makes delay guarantees impossible whether done at the optical layer or by TCP. Adopting a two-way reservation approach [2, 3] solves the loss problem but the reservation delay is only acceptable for networks with a circumference of 1000–2000 km.

Also, the efficiency of the reservations increases with higher burst aggregations, but this adds to delay.

Given the very different strengths and weaknesses of optical switching [4] compared with the electrical one (limited and costly buffering and processing, but many WDM channels and no electrical/optical payload conversion) a departure from traditional architectures towards one that exploits the strong points of optical technology, while side-stepping its disadvantages, makes sense. The novel hybrid architecture proposed in this paper, combines two-way reservations inside clusters of nodes in relative proximity, with one-way transport outside such clusters via provisioned light paths or burst switching. This way it combines the strong points of both while alleviating their drawbacks. (It is worth mentioning that clustering has also been proposed in Reference [5], however only to aggregate

*Correspondence to: H. C. Leligou, National Technical University of Athens, School of Electrical and Computer Engineering, Telecommunications Systems Laboratory, 9 Heroon Polytechniou Street, Zographou, 15773, Athens, Greece. E-mail: nelly@telecom.ntua.gr

Received 3 March 2006
Revised 12 January 2007
Accepted 17 September 2007

Copyright © 2007 John Wiley & Sons, Ltd.
traffic for improved light path utilisation without dynamic sub-wavelength allocations as proposed here.)

Another angle to approach the concept is to make a parallel to early LANs, where the heavy cost of switches led to a shared medium approach, where stations act like ports of a distributed switch sending and picking own packets under proper access control. When later cheap VLSI chips made frame switching cost-effective centralised switching was introduced. Today, while components capable of fast optical switching are quite expensive, the inherently fast response of fibre-optic transmitters can be exploited in a ring to create trains of bursts with the same destination under the coordination of a reservation protocol in a fashion similar to Reference [6]. This motivated us to propose a similar distributed switching technique for the all-optical core network featuring high multiplexing gain, reducing the prohibitive loss of on-the-fly bufferless burst switching.

2. PRINCIPLE OF OPERATION

Given a set of core nodes as in Figure 1, neighbouring nodes are grouped to create clusters inside which nodes are connected to form a ring (probably along the ducts of already existing SDH ring). A core network of 20–30 nodes, with its longest round-trip time at about 50 ms, can be broken into 4–5 clusters of 5–6 nodes with a round trip time of about 5–10 ms. As shown in Figure 1, one node in each cluster ring, the Master Node (MN), hosts the reservation protocol and is the gateway to other clusters. The MNs are connected via an inter-cluster network of provisioned or dynamic light paths.

The periphery nodes possess electrical buffers and use separate queues per destination cluster as they create fixed size slots (with a size in the order of a fraction of an ms) from arriving IP packets. The use of slotting allows more efficient control since reports and allocations can refer to large chunks of data. Slot headers include the destination node to allow reception inside the destination cluster. Slots are aggregated on the fly into larger frames with the same destination cluster by forming composite bursts (frames) under the guidance of the reservation protocol as will be described below. These frames are then forwarded also on the fly from one MN to another via the provisioned inter-cluster network. The frames can be of fixed or variable lengths (but still an integer multiple of slots).

The reservations inside each cluster that create the burst aggregations work as follows:

One of the WDM channels is devoted to control information which is carried in fixed size control frames. The
control frame is of the order of the round trip time and includes two kinds of information in its fields: reservations in the Slot Reservation Map (SRM) section and allocations in the Slot Allocation Map (SAM) section as are shown in Figure 2. The SRM is a periodic field allowing the MN to collect requests in the form of Queue Length (QL) information (reservations) from all nodes giving the number of slots queued per destination cluster (cf with MAC protocols for PONs e.g. [7]). On the basis of this information, the MN can create a mirror of the queue situation in all cluster nodes and assign slots in the control channel. Each node contributes the allocated number of slots into the data channels creating a contiguous train of slots all destined for the same cluster. The end result is a system with the ability to respond to traffic fluctuations, while creating aggregations that can be efficiently directed to the destination cluster, where each node will pick its own traffic.

The control channel, which is the only one to be converted to the electrical domain in each node, is organised in fixed frames by the MN (Figure 2). The control frame starts with the Frame Alignment Pattern, (FAP). The SAM follows based on earlier reservations and then the SRM for the next collection of node requests. In the SRM, the MN marks the destination cluster address, followed by the successive node addresses, leaving empty fields where each node adds, as it passes by, its queue lengths (per destination cluster and QoS class) and the error control field. When the control returns to the MN, a snapshot of all the waiting traffic for all C clusters becomes known to the MN, though slight changes have occurred since, but this is no problem in the longer run (cf [6, 7]).

Regarding the inter-cluster network among MNs, this can start as a full mesh of static provisioned light paths with multiplexing gain only inside the cluster. This is simple and cost-effective for 4–6 clusters. As the network grows, the poor scalability of a full mesh can be side-stepped by adopting a more dynamic automatically switched optical network to set up the light-paths among MNs according to demand. The burst packing and reduction of source–destination pairs to clusters rather than individual nodes helps efficiency in this case as well. A completely dynamic inter-cluster network is also possible by using one-way OBS among MNs at the penalty of introducing contentions at an intermediate or destination MN. Adopting fixed frames can reduce the speed and cost of intermediate space switches. Still, the loss probability would be much reduced (only one contention point) compared to the joint probability of successfully crossing 3–4 contention points if all the nodes were connected via OBS instead of the much fewer MNs.

A further benefit of more dynamic inter-cluster solutions is the improvement in the ability of the system to handle efficiently asymmetries in traffic among clusters compared with the static provisioning which does not allow the operator to handle fractions of a whole light path in response to traffic asymmetries between clusters.

Copyright © 2007 John Wiley & Sons, Ltd.

DOI:10.1002/ett
3. THE ALLOCATION ALGORITHM

As reservation information arrives periodically in the SRM, the MN continuously updates a reservation matrix $R$, where element $r_{ij}$ contains the number of slots wishing to go from node $i$ to cluster $j$. To support two QoS classes, two such matrices should be used: one $R_H$ for high priority traffic and another $R_L$ for low priority traffic, each based on information from the respective $QL_H$ and $QL_L$ fields of the reservation map. The MN also maintains an array $S$ where each element $s_i$ contains the total number of slots destined for cluster $i$ calculated by adding the arriving queue lengths from all nodes for this cluster including both priority classes. In order to enforce upper limits to the delay of the system, the MN maintains a set of timers which measure the time between the announcement of a request and its service, i.e. the time it is decided to form a frame towards the requested destination cluster. The above information is used to prepare the SAM, in the first section of the control frame shown in Figure 2, which is sent ahead of the corresponding payload frames in the other wavelengths by enough time to allow the nodes to write in the relevant positions as indicated by the control allocations. The SAM is organised in $w$ fields, each describing the Destination Cluster Address (DCA), i.e. the address of the cluster to which the frame under formation will be eventually sent when it arrives at the MN, and the allocations of slots to individual nodes for a certain wavelength. The allocations to each node provide the exact position in the frame by means of pointers indicating the starting and finishing slots, as depicted in the figure. The node will insert in the control field DA (which is left empty by the MN) the actual destination node address, which will be used by the receiving node in the other end to identify which slots are addressed to it. So, the Error Correction (EC) field needs to be recalculated.

The MN prepares the allocation map using a simple algorithm, amenable to easy implementation of its operations by dedicated H/W logic in one slot time. The output of the algorithm is the destination of the next frame and how many slots of the frame go to each node and both are decided on the basis of the information in the array $S$ (total slots for both priorities), the reservation matrices $R_H$, $R_L$ and a set of timers giving the upper bound of the frame repetition period for each class of service per destination cluster. The timers measure the time from the first non-zero request announcement towards a destination cluster until the allocation of a frame destined to this cluster is decided; they are reset when the total number of announced slots towards this cluster is satisfied. In any case, the timers do not run on empty reservation totals to a destination cluster. The two QoS classes are mainly differentiated by the threshold value of the timers and the fact that high priority can ‘steal’ the positions of low priority even if low priority caused the reservations in the first place. The node can use the allocated slots at will regarding the QoS of the payload, i.e. will insert high priority slots irrespective of what queued slots triggered the reservations that resulted in the current allocations. Such slots that depart will not be of course included in the requests (unless already done) while low priority slots that still remain in the queue will again be reported in the reservations ($QL_L$ field).

The main steps of the algorithm are as follows ($L$ is the frame size measured in slots):

- Decide frame allocations: A frame destined to cluster $i$ will be formed, if $S[i] \geq L$ or $T_H = 0$ or $T_L = 0$, where $T_H$ and $T_L$ denote the high and low priority timers for destination cluster $i$.
- Decide slot allocations to nodes: if $S[i] < L$ then grant all nodes’ requests else.
- If $\sum_j r_{ij} \geq L$, grant high priority requests proportionally to service weights.
- If $\sum_j r_{ij} < L$, grant all high priority requests and then grant low priority requests proportionally to service weights.

The weights, which sum up to 100%, represent the fraction of traffic that the core operator has admitted from each node. They are calculated on the basis of Service Level Agreements and express the traffic ratio that each node is entitled to take from the total ring capacity. The pro-rata allocation according to these weights at momentary overloads (prolonged ones are precluded by admission control and policing at the ingress) constitutes at the same time the fairness mechanism that is necessary in every ring. The ideal weighted allocation cannot be respected in every
frame but residual ‘debts’ are covered over the next frames satisfying the perfect fair state over longer timescales. Slotting facilitates the bookkeeping related to fairness but also the implementation in general.

Allocations must be updated with every frame and consequently a H/W implementation is mandatory, assisted by an embedded controller for configuration support and management (e.g. weights, wavelength allocations, lightpath updates etc.). The algorithm described in Reference [6] for a metropolitan ring, which is of similar complexity has been implemented in an FPGA featuring a pipelined execution in three slots of 1 µs duration as described in Reference [8]. In this case, much longer slots (e.g. 100 µs) are used presenting no implementation difficulty in spite of the higher core rates.

4. PERFORMANCE EVALUATION

To assess the performance, an event driven simulation model was prepared. The model comprised five cluster rings of 5 ms round trip time (RTT), with six nodes in each cluster. Fixed-size frames equal to the RTT were used consisting of 50 slots, i.e. 0.1 ms per slot. Four payload wavelengths were used in each cluster, while between the clusters each λ was provisioned to one of the other clusters with an inter-cluster link of 1000 km length (10 ms). The rate at each λ was 10 Gbps. Both Poisson and Self-similar sources were simulated to generate the offered load, which was uniformly generated by all nodes and destined to all nodes and clusters with equal probability. Self-similar traffic in each node consists of 20 ON–OFF sources per destination cluster (per queue) generating slots with Pareto distribution for both the ON and OFF durations (Pareto shape 1.3). The high priority traffic was 30% of the total. The timer values were 25 ms for high priority and 300 ms for low. The latter is seldom used to force a frame since low priority traffic usually rides frames initiated by high priority time-outs or by exceeding the frame length.

Figure 3a shows the average end-to-end delay against the total offered load. This is the queuing delay plus on average 15 ms propagation delay (i.e. 2.5 to traverse the ring, 10 ms to travel from MN to MN, and another 2.5 to traverse the other ring). It is remarkable that the delay of the high priority goes below even the reservation delay, decreases with the offered load and is not affected by the higher burstiness of self-similar sources. All three effects are explained by the fact that high priority traffic does not have to wait for allocations caused by their own reservations, but can ‘steal’ those allocated to low priority reservations. In other words, it occasionally enjoys immediate access to slots incidentally passing in front of the node saving the reservation mechanism as a contingency plan in the event that no lower priority traffic is also present. Its advantage is such that high priority traffic is not affected by even self-similar sources, which naturally bring earlier to instability the low priority.

Traffic prioritisation also allows the high priority class to enjoy good delay performance even above 100% load, since all the instability is suffered by the low priority (which will eventually reduce its load via closed loop mechanisms at transport layer (e.g. TCP)). Overall the system can take up to almost 90% loading which proves the high efficiency of the approach compared with OBS.

![Figure 3](image-url)

Figure 3. (a) End-to-end delay vs. load (b) PDF of queuing delay at 80% Poisson load.
It is worth commenting that the delay is higher at low loads because most frames are generated due to timer expiries and not due to filled frames. The fill levels of the produced frames become almost 100% full for offered loads above 30%. This means that all the gaps are aggregated as inter-frame space allowing for efficient sub-lambda switching in the inter-cluster network if such an approach is adopted among MNs.

The Probability Density Function (PDF) of the queuing delay at an 80% Poisson load is shown in Figure 3b. For high priority traffic, the framing dominates, explaining the step-wise shape of the PDF. Most traffic departs within one RTT while some go for the second, with few that encountered momentary congestion needing a third. Higher values in a bell-shaped form are exhibited by the low priority traffic, but still delay performance is quite satisfactory for the elastic services they represent. The pdf shows that most high priority traffic departs before the 5 ms reservation delay in frames inititated by other traffic, thus enjoying excellent performance suitable for real-time services.

5. CONCLUSIONS

By allowing the control logic to arrange the multiplexing of traffic destined to the same clustered group of nodes, the same effect as would be created by expensive fast all-optical switching is achieved. This hybrid architecture (two-way reservations inside the short distances of a cluster, one-way transport for the longer rest of the journey) results in efficient aggregation and multiplexing gain without transit loss and excessive delay. Thus, clustering exhibits neither the intolerable delay of two-way reservation approaches that attempt end-to-end reservations by signaling protocols, nor the intolerable inefficiency and losses of one-way systems, due to output port contentions. The proposed solution mimics early switch less LANs where the high switch cost led to adopting a distributed version executed by the nodes acting as ports with the MAC protocol in the role of the switch controller.

ACKNOWLEDGEMENTS

This work was funded by the EU IP project FP6-506760 ‘NOBEL’.

REFERENCES