Hybrid burst/packet switching architectures from IP NOBEL

Helen C. Leligou*a, Gert Eilenbergerb, Lars Dembeckb, Wolfram Lautenschlaegerb, Stephan Bunseb, A. Stavdasd, J. Angelopoulosa, Christina (T.) Politic

*aICCS-National Technical Univ. of Athens, Heroon Polytechniou 9, 15773 Athens, Greece; 
bALCATEL SEL AG, R&I, Holderaeckerstr. 35, 70499 Stuttgart, Germany
cDept. of Telecommunications Science and Technology, Univ. of Peloponnese, Karaiskaki str., 22100, Tripoli, Greece

ABSTRACT

In spite of its long term promise, all-optical switching is still plagued by high cost, low efficiency when handling bursty data traffic, immature management and protection and poor output port contention resolution leading to heavy loss. Given the current situation, hybrid approaches that keep the best features of optics, reverting to the electrical plane when expedient, constitute sensible interim steps that can offer cost-effective solutions along the road to an eventual all-optical core. Two such approaches developed in the framework of the European IP project NOBEL are presented in this work. The first is a quite mature solution that extends present day concepts to achieve multiplexing gain while keeping all the management and restoration benefits of SDH. The other mimics early LANs in executing a distributed switching via its electrical control plane using two-way reservations, thus restricting its applicability to smaller domains. Combining the two leads to a system fulfilling most of today’s requirements for Tb/s core networks.

Keywords: Optical networks, Optical Burst Switching, OBS, optical switching

1. INTRODUCTION

The circuit-switching paradigm has reached its limits, failing to provide efficient support of modern data-centric packet-based services despite useful patch-up extensions, such as LCAS and Virtual concatenation that have added some flexibility to SDH networks. Core networks are in need of more dynamic architectures that can exploit advances in photonics and offer efficient multiplexing of traffic with varying needs. The introduction of the optical burst-switching approach [1], [2] attempts to enrich WDM technology with statistical multiplexing properties in the quest for efficient accommodation of the temporal and/or spatial fluctuations of modern traffic without heavy channel over-provisioning as in wavelength-routed networks. However, the significant shortcomings of this approach hinder wide experimentation and hold little promise of commercial deployment unless further improvements are realized. The costly and unwieldy optical buffering leaves such systems exposed to heavy loss [3], [4] because of contention, leading in turn to low efficiency in an effort to reduce collisions by low channel filling.

To improve on these problems, hybrid solutions have been investigated in the framework of the EU project FP6-506760 “NOBEL” and two of them are presented in this work. The first one keeps whatever is mature in the optical part while it seeks the benefits of the proven and low cost elements from the electrical part. Packets are aggregated in frames that are handled with multiplexing gain and flexibility, though the G.709 container is kept to keep disruption low since electrical buffering and EO conversion can be used at least in the initial stages. A migration path towards an all-optical solution is envisaged that seeks to encompass future optical components as they become available replacing electrical parts once they are surpassed in performance.

The second approach is another kind of hybrid: combining two-way reservations inside geographically limited domains with one-way transport among domains, such as semi-static provisioned light-paths or by means of the previous approach allowing virtually unlimited network sizes and traffic rates. Thus, inside each domain, which is a ring formed by a cluster of core nodes, loss is eliminated, while traffic from many nodes can be aggregated into single bursts improving efficiency. Clustered nodes contribute contiguous optical slots, which are marshaled into larger optical composite containers destined for nodes in the same destination cluster, without any collision or loss under the guidance of control information from a reservation-based MAC protocol.

*nelly@telecom.ntua.gr; phone +30-2107721511; fax +30-2107722534; ong.ece.ntua.gr
2. LAYER 2 FRAME SWITCHING

The layer 2 frame switching architecture (L2FSA) is, in contrast to pure optical burst switching, a hybrid optical / electrical solution. It combines the aggregation into large containers as featured in OBS with the convenient queuing and scheduling afforded by electrical memory buffers. Its switching granularity is well adapted to the features of optical space switches, where the switching speed is typically lower than the bit rate by several orders of magnitude, whereas in the electrical domain bit rate and switching speed are almost identical.

The architecture is intended to be a convergence technology for telecommunication core networks. It has to cover all kinds of services present in networks today in the sense that the same uniform equipment should be able to transport any mix of those services. The mediation between services, however, is not regarded as a task of the L2FSA architecture. Any service that enters the L2FSA network leaves the network as it is, making L2FSA a pure transport network leaving service associations to the edge device.

Ethernet and MPLS as common layer 2 protocols have the disadvantage of too small packet sizes, and on top of this they are also variable requiring many forwarding decisions per second from the switch stressing its control functions and cost. Here a fixed-size composite packet (frame) is employed, and as example the G.709 frame format could be adopted, where every G.709 frame constitutes a L2FSA PDU. Like other OBS technologies, L2FSA also allows for a decoupling of the speed of switching and control from the ever-increasing link capacity. The packet size could grow with the evolution of link capacity while the switching times remain stable.

Another quite important advantage relates to the synchronization of the high-speed core networks. With L2FSA it becomes possible to de-couple clock domains enabling an asynchronous network operation.

2.1 The L2FSA Network and Node Architecture

Here the main features of the L2FSA concept will be discussed. First, any kind of client traffic is aggregated into large, equally sized containers or frames. As mentioned, an initial proposal for the container could be the G.709 Optical channel Transport Unit (OTU; 16320bytes). The aggregation is done separately per destination and service class, so that packets can be switched and forwarded throughout the whole network as indivisible entities. Therefore, frames are routed along predefined virtual connections, the so called Label Switched Paths (LSP). In a meshed core network such LSPs are typically set up only for connections with reasonable traffic demand. The lower bound for an LSP is roughly 100Mb/s in a core network of 10Gb/s links.

Traffic is classified into two or maximum three quality of service classes. In case of resource conflicts any higher quality class has strict priority over a lower quality class. Complementary, for fairness reasons, the higher quality classes are strictly bandwidth limited, so that any link is loaded by less than 50% with higher quality classes.

Links between the core nodes are state-of-the-art continuous transmission protocols, preferably the G.709 protocol. Frames are embedded into the continuous bit stream (logical frames).

In figure 1 a network scenario is shown with IP/MPLS routers providing the client traffic and a L2FSA in-between for realizing the connectivity among the IP routers. The MPLS paths start and end in the client layer network. There is no interaction foreseen between the MPLS router paths and the L2FSA network. The frame aggregation unit (FAU), which performs the aggregation of the user traffic in a stream of frames, executes a 1:1 translation of a router port onto the frame switch port. In order to allow for QoS handling, a separate frame assembly is performed for every path and service class.

Figure 2 shows a frame switching node in detail. The upper part depicts the frame switch, which has adapted G.709 OTUx interfaces. The frame switch part consist of line cards and an internal matrix (not shown in figure 2) The frames are stored on input line cards and then forwarded individually to the desired output line card (packet switching). On output line cards a new continuous G.709 link connection is assembled from the received frames, possibly by addition of empty / filler frames. For contention resolution virtual output queuing is applied. The matrix operates frame synchronous but asynchronous in terms of bit clock. Bit synchronization is recovered in the outgoing line card. This does not only ease the synchronization but also allows for a simply application of an optical matrix.
The frame aggregation unit (FAU) located below the frame switching part takes the data from the client interfaces and forms the containers for transport in the frame switching network. In the reverse direction the frame stream is terminated and the client traffic is lead to the outgoing client interfaces.

**2.2 QoS handling in the Frame Switch concept**

In a first step three different QoS classes can be defined. Later on, there can be more, but more classes always mean more operational effort and the possibility to consolidate traffic decreases. The first class contains TDM services with a constant bit rate. The second one includes services with a controlled and guaranteed bandwidth. VoIP is a typical representative of this class but also most VPN services will wish to have such a kind of quality. Unlike class 1, in this class statistical multiplexing can be achieved. The third class contains the best effort service, covering all services offered in today’s Internet.

There is a one by one mapping of the QoS of the client classes and the QoS of the frames. In order to avoid any influencing of lower class traffic on the properties of higher class frame streams even the possibility to fill up high priority frames with low priority payload was not allowed.

The highest priority class “TDM traffic” is assigned to a calendar type time schedule meaning a periodic meta-frame on top of the frame structure (like a week is a structure on top of the sequence of days). In opposite to other scheduling schemas the calendar setting for a path is fixed over the path’s lifetime. When a path through the network is set up, all nodes on the path check their own calendar, whether or not this traffic can be served. If this is the case, the node’s scheduler reserves a periodic time slot for that particular path.
The TDM traffic class is intended for continuous bit streams like SDH tributaries. It is for legacy compliance only, where traffic sources do not take care of the burst/packet nature of the network. The policy is also applicable to bursty packet sources with exceptional safety needs. The TDM traffic class is assumed to be less than 10% of the total traffic volume in future.

Second priority is “Guaranteed Bandwidth Traffic”. Each frame of this class is followed by a mandatory waiting time. After the expiration of the mandatory waiting time the FAU is free to emit a frame at any time slot. The mandatory waiting time can be a fractional multiple of the slot time. In this case, the compliance with the mandatory waiting time is taken in a cumulative way. Example: waiting time 1.5 yields a sequence of inter frame gaps like this: 1-2-1-2-1-

When a path is set up through the network, each node checks, whether or not the requested guaranteed bandwidth can be served in context of the total channel occupation. There is no guarantee for an individual frame to catch a free time slot at intermediate nodes. In case of contention, the frame is queued in a first-in-first-out policy among all competing frames of the same class.

A third traffic class is Best Effort: It is unlimited in all dimensions but in case of contention it has always the lowest priority (no fairness with other classes). However, due to the hard limitations for the higher traffic classes, some bandwidth for best effort is always available. It is a question of buffer size and permitted waiting time to control the frame loss rate up to a certain channel occupation.

### 3. CANON ARCHITECTURE

The CANON architecture is a novel hybrid architecture that 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. The concept is similar to early LANs, where the heavy cost of switches led to a shared medium multicasting approach that let each station send and pick their own packets carrying out an implicit distributed switching action by proper access control. When later cheap VLSI chips made frame switching cost-effective centralized switching was introduced. Today, while components capable of fast optical switching are quite expensive, the inherently fast response of fiber-optic transmitters can be exploited in a ring to create trains of bursts with the same destination. Adopting a two-way reservation approach [5], [6] solves the loss problem but the reservation delay is only acceptable for networks with a circumference of 1000-2000km. Thus under the coordination by a reservation protocol in a fashion similar to [7], a packet multiplexing can be produced simulating the operation of an output port of a fast switch, without centralized switches. The above considerations motivated us to propose a similar distributed switching technique for the all-optical core network featuring high multiplexing gain, without the prohibitive loss of on-the-fly optical switching.

#### 3.1 Architecture

Given a set of core nodes as in Figure 3, neighboring 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 50ms, can be broken into 4-5 clusters of 5-6 nodes with a round trip time of about 5-10ms. As shown in Figure 3, one node in each cluster ring, the MN (Master Node), hosts the reservation protocol and is the gateway to the other clusters. The MNs are connected via a network of provisioned or dynamic lightpaths forming a full or partial mesh, or even a ring with provisions for redundancy and robustness.

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 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 and fields to recreate IP packets on the other end of the core. Slots however, are aggregated on the fly into larger composite containers with the same destination cluster under the guidance of the reservation protocol as will be described below. These containers are then forwarded also on the fly from one MN to another via the inter-cluster network. These composite containers can be of fixed or variable length (but still an integer multiple of slots). Fixed size containers are adequate and simpler to handle since at a first stage the multiplexing gain inside the cluster provides enough efficiency and (though perfectly possible) there is little incentive to pursue it in the inter-cluster network as well.
Fig. 3 An example of clustered network and principle of operation

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 containers. The control container 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 4. 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. 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 organized as shown in Figure 4. The control container starts with the container alignment pattern, (CAP) which is used for physical layer bit/byte alignment purposes. Then follows the SAM based on earlier reservations and then the SRM for the collection of nodes 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 (for the destination cluster in question and each QoS class) and the error control field. Thus, after a control frame time all the waiting traffic for all C clusters become known to the MN.

3.2 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 composite container towards the requested destination cluster. The allocations are also checked against service-based weights expressing Service Level Agreements (SLAs).

The above information is used to prepare the Slot Allocation Map (SAM), in the first section of the control container shown in Figure 4, which is sent ahead of the corresponding payload containers 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 organized in w fields, each describing the Destination Cluster Address (DCA), i.e. the address of the cluster to which the composite container 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 container 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 container 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 (QLL field).

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

- Decide composite container allocations (i.e. destination cluster) : A composite container destined to cluster \( i \) will be formed, if \( S[i] \geq L \) or \( T_{H,i} = 0 \) or \( T_{L,i} = 0 \), where \( T_{H,i} \) and \( T_{L,i} \) 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 \( \sum_{j} R_{H}[j,i] \geq L \), grant high priority requests proportionally to service weights
  
  - if \( \sum_{j} R_{H}[j,i] < L \), grant all high priority requests and then grant low priority requests proportionally to service weights

It is worth noting that, when all queues are always busy (backlogged system) the timers dominate the allocation and the algorithm ends up providing round robin allocations to the destination clusters according to the provisioned shares (SLA-based weights), which determine the value of the timers.
4. PERFORMANCE EVALUATION

4.1 L2FSA Performance Evaluation

The policy for Guaranteed Bandwidth traffic as explained in chapter 2.2 is the key factor for proper operation of the scheduling algorithm.

In short, the bandwidth for Guaranteed Bandwidth traffic is allocated in every intermediate node during the setup phase (setup of label, routing table entries etc. for one particular path through the network). A path cannot be set up as a Guaranteed Bandwidth path, if at an intermediate link the cumulative sum of all allocated Guaranteed Bandwidth paths exceeds a certain limit, e.g. 30% of the link capacity. At the ingress of the network, the frame aggregation unit (FAU) keeps track of this bandwidth allocation. After emission of a frame the FAU waits for a certain time interval according to the granted bandwidth. After expiration of the wait time the next frame can be sent at any time. However, times that are lazed away cannot be regained.

This strong policy allows a flexible statistical filling of the path below the bandwidth limit, but it avoids the unpredictable “heavy tailed distribution” problem of ordinary uncontrolled internet traffic. At the bandwidth limit a Guaranteed Bandwidth path behaves much like a TDM path. The guaranteed wait time prevents buffers in the next node from overrun due to temporary traffic peaks. Furthermore the strong limitation for Guaranteed Bandwidth traffic to ~30% of the link capacity ensures a permanent availability of bandwidth for the best effort traffic service, since it is not desirable to totally block Best Effort traffic during peaks of the Guaranteed Bandwidth traffic.

Due to the required buffering for contention resolution, the statistical features of the Guaranteed Bandwidth traffic degrade after each intermediate node. Even though the average bandwidth remains within the granted limits, the probability for traffic peaks increases. A common mitigation to this problem is an additional buffering for traffic shaping. This equalizes traffic peaks and avoids uninterrupted chunks of frames, which would otherwise block Best Effort traffic for an unpredictable time. In the sequence, we try to quantify the statistical effects.

In our analysis we assume the following conditions:

We consider a number of \( N \) flows with a total bandwidth sum \( B \) (frames per second), that follow the policy of Guaranteed Bandwidth traffic as described above.

These flows that operate at the bandwidth limit are randomized to a certain degree by traffic shaping. (Without this rule their frame rate would be deterministic and constant and with unlimited correlation span.)

Then we can state the following hypothesis for the merged flow:

- For small \( N \) (low number of thick flows) the merged flow behaves not worse (in terms of buffer need) than a frame flow at constant rate \( B \).
- For large \( N \) (large number of thin flows) the merged flow behaves not worse (in terms of buffer need) than a Poisson flow at average rate \( B \).

The treatment of first hypothesis (small number of flows) is trivial: For a constant flow, the buffer need is zero as long as the rate is below 100% of the channel capacity. Taking into account the randomization by traffic shaping the buffer need grows up slightly below the 100% limit. However, at the envisaged maximum share of 30% for Guaranteed Bandwidth traffic the buffer need will be still close to zero.

The second hypothesis (large number of flows) leads to a queuing problem with random uncorrelated arrival but deterministic service (M/D/1). In our scenario the output port can serve one frame per slot, but at the same time up to \( N \) frames could target that port. That means, if \( k \) frames arrive, then \( k-1 \) frames need to be stored in the output queues. In a steady state of the queues with, e.g., \( m_{i-1} \) frames already waiting and \( k_i \) new frames arriving the new total queue length \( m_i \) is:

\[
m_i = m_{i-1} + k_i - 1
\]  

(1)

However, if the queues were empty before (\( m_{i-1} = 0 \)) and there is no packet arrival (\( k_i = 0 \)) then the bandwidth is wasted and the queues remain empty (\( m_i = 0 \)),

(2)
On the other hand, if the new $m_t$ becomes larger than the installed buffer space $M$, then $m_{t-1} + k_t - 1 - M$ frames get lost and the queue remains at maximum $m_t = M$. Figure 5 illustrates the model.

$$k_t$$ frames arrive at time $t$

0 or 1 frame can leave per slot

max $M$ buffer cells

For large port number $N$ the probability to receive $k$ frames with the same target port at the same time is:

$$P(k) = \frac{B_r^k e^{-B_r}}{k!}$$

where $B_r$ is the relative output channel occupation, reserved for the Guaranteed Bandwidth traffic (sum of all flows targeting that particular output port).

Thus, the transition from buffer filling $m_{t-1}$ to $m_t$ is a stochastic process depending on the probabilities $P(k)$ and the state transition rules 1-3 as above. The particular buffer fillings $m_t$ constitute a discrete-time Markov chain. The solution of the corresponding balance equations is given in [8], from where the frame loss probability can be derived. Figure 6 shows the frame loss probability depending on the maximum queue length $M$ for different levels of channel occupation $B_r$.

As we can see a channel occupation $B_r = 0.3$ yields frame loss probabilities less than $10^{-9}$ at buffer size $M = 10$. So, together with the hypotheses above we expect that Guaranteed Bandwidth traffic can be controlled in a way where all frames are reliably delivered.

As an example a 10Gb/s line card with 100 different LSP’s (remember the minimum bandwidth of a LSP is set to about 100Mb) and three service classes would require a buffer space of $10 \text{ frames/queue} \times 100 \text{ destination ports} \times 3 \text{ service classes} = 3000 \text{ frames or 50MB} \text{Byte memory per line card for virtual output queuing. This is well within the current technological possibilities; IP routers normally have at least a ten fold higher buffer space on their 10Gb/s linecards.}

![Fig. 6 Frame loss probability as function of buffer size (from [5], extended to low $B_r$)](image-url)
4.2 CANON performance evaluation

To assess the performance of the CANON architecture, an event driven simulation model was prepared. The model comprised 5 cluster rings of 5ms round trip time (RTT), with 6 nodes in each cluster. Fixed-size composite containers equal to the RTT were used consisting of 50 slots, i.e. 0.1ms per slot. Four payload wavelengths were used in each cluster, while between the clusters each wavelength was provisioned to one of the other clusters with an inter-cluster link of 1000km length (10ms). The rate at each wavelength was 10Gb/s. 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) with Pareto distribution for both the ON and OFF durations (Pareto shape 1.3) and a burstiness of 10. The high priority traffic was 30% of the total. The timer values were 25ms for high priority and 300ms for low. The latter is seldom used to force a composite container since low priority traffic usually rides composite containers initiated by high priority time-outs or by exceeding the container length.

Figure 7a shows the average end-to-end delay against the total offered load. This is the queuing delay plus on average 15ms propagation delay (i.e., 2.5 to traverse the ring, 10ms to travel from MN to MN, and another 2.5 to traverse the other ring). Two remarkable observations are worth making on the delay of the high priority: it is about 10ms lower than that of the low priority (even at very low loads when all requests are served) and decreases with the offered load. Both 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. Traffic prioritization 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. The difference between Poisson and Self-similar traffic manifests itself in the average delay at the earlier instability for the latter.

It is worth commenting that the delay is higher at low loads because most composite containers are generated due to timer expiries. It is obviously easy to reduce this delay by a contiguous generation of composite containers, at no penalty for the inter-cluster efficiency as long as the whole wavelength is provisioned to one destination cluster. However, in future work we plan to investigate more dynamic inter-cluster networks with one-way optical switching, and this needs a mechanism that lowers link occupancy in proportion to the offered load. The performance results show that the fill levels of the produced composite containers are almost 100% full for offered loads above 30%. This means that all the gaps are aggregated as inter-container space allowing for efficient sub-lambda switching between clusters.

The probability density function (PDF) of the queuing delay at an 80% Poisson load is shown in Figure 7b. 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 in quantitative terms that most high priority traffic departs before the 5ms reservation delay in composite containers initiated by other traffic, thus enjoying excellent performance.
5. COMBINING THE TWO APPROACHES

The two architectures were presented and evaluated separately since they have not been integrated so far. However, they possess complementary features and the objective is to be combined into an architecture that combines the best features of both being unlimited by the delay of the two-way reservations of CANON, and avoiding the low fill levels of the L2FSA approach. To this end, rings of nodes that employ the CANON control plane and posses the FAUs of L2FSA are connected in an arbitrary partial mesh via MNs that can switch L2 frames. In other words, it is a L2FSA system that collects frames via a distributed set of FAUs controlled by the two-way reservations of the CANON architecture. Thus CANON is used inside the cluster for the collection of G709 frames with the same cluster destination in a distributed fashion in larger fixed duration (in the order of few ms) layer 2 frames while L2FSA is used between the clusters. Separate reservations and frames are used per QoS class allowing prioritized assignment of high class traffic inside the cluster but are prioritized switched in the inter-cluster L2FSA switches of the mesh. This combined architecture allows the coverage of areas the size of Europe with rings covering small countries (multiple ones for bigger countries) interconnected via L2FSA nodes that have no distance limitations. The end result is a system that combines the efficiency of reservations with the flexibility and range of frame switching. Further work is required to reach a smooth integration of the two concepts.

![Diagram of the combined architecture](image)

6. CONCLUSIONS

To enrich core networks with more dynamic architectures that exploit advances in photonics to offer efficient multiplexing of traffic, two hybrid solutions with complementary features have been presented. They keep the mature features of optical parts while seeking to exploit the benefits of the proven and low cost elements from the electrical part. The one is based on two-way reservations and offers efficiency albeit for limited size networks, while the other is unfettered by reservations and can extend to any network dimension. The combination of the two, promises a powerful solution that offers efficiency, flexibility and robustness in near term deployment. A migration path towards an all-optical solution is envisaged that seeks to encompass future optical components as they become available replacing electrical parts once they are surpassed in performance.

ACKNOWLEDGEMENT

This work was funded by the EU project FP6-506760 “NOBEL”. The views expressed herein are those of the authors and do not necessarily represent the position of the whole NOBEL consortium.

REFERENCES