

Light-Mesh: An Evolutionary Approach to Optical Packet Transport in Access Networks

¹Ashwin Gumaste, ¹Tamal Das, ¹Raviraj Vaishaympayan and ²Nasir Ghani

¹Dept. of CSE, Indian Institute of Technology, Bombay, Powai, Mumbai, India, 400, 0076.

²Dept of ECE, University of New Mexico, Albuquerque, NM, USA, 87131.

Email: ashwing@ieee.org, tamal.das@cse.iitb.ac.in, raviraj@cse.iitb.ac.in and nghani@ece.umn.edu

Abstract: We report achieving optical packet transport using mature off-the-shelf components as an alternative technology to PON called *light-mesh*. Novel service provisioning algorithm and implementation details are discussed with delay/utilization profiles presented.

© 2008 Optical Society of America

OCIS codes: 060.4250 Networks; 060.4250 Networks

1. Introduction and Network Philosophy

We present for the first time an implementation of optical packet transport as a solution for the access area. Two features affect design of optical access networks: (1) Traffic being characterized as – packetized (sub-wavelength granularity per node), delay sensitive, multi-service oriented. (2) Fiber deployment contributing as high as 90 % of network-layout cost. We argue a case for new network architecture in the access and have in [1] shown the same to be lower cost than star-PON (see Fig. 1 and Fig. 2). This architecture based on the principles of a packetized-mesh (with optical packet transport), that we proposed in [1-3] and called *light-mesh* has been shown to result in 70%~85% capital expenditure savings) than contemporary star-PONs (see Fig. 1 and Fig. 2. In a traditional mesh, nodes require active (switching) elements which are difficult to operate and provision in the consumer-centric access area. By using a set of *passive* mesh nodes with select *active* nodes

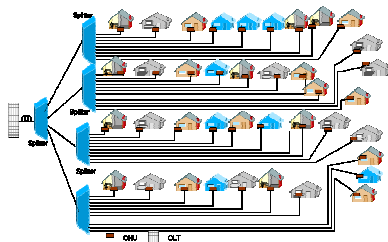


Fig. 1. Conventional PON [3].

it is possible to create a mesh network that would result in net fiber savings and better per-meter utilization of fiber. The principle challenge in deploying nodes in mesh configuration with passive and low-cost features is to enable packet oriented quality-sensitive communication. We showcase the light-mesh test-bed that supports packet transport and triple-play traffic.

Network philosophy: The light-mesh consists of two types of node interconnections and two types of nodes. The node interconnections are called *strings* and *threads*. Strings are single-wavelength optical buses that are time-shared by multiple nodes. Threads are point-to-point optical connections between two nodes in the network. The strings and threads are assigned in a way that together they lead to N^2 connectivity within the light-mesh, i.e. there is no need for intelligent routing/address recognition or optical storage; the physical layer guarantees end-to-end transport as explained in [1]. All nodes within a string are passive nodes (*OOO*), while the start and end-nodes of a string (that interconnect to other strings/threads) are active nodes (*OEO*). Active nodes are those that convert optical signal (and hence packets) to electronic signal, do a layer-2 address match, and then re-transmit (packets) subject to a topology discovery algorithm that determines if a packet is to be forwarded or not. Passive nodes are simply those that support a bus, i.e. using drop-and-continue and passive add functionalities. Communication within the light-mesh happens by each node sending in packets (called light-frames – LF [1]), which are Ethernet frames appended with VLAN tags. A node sends in LFs using principles of native Ethernet (CSMA), with CSMA implementation based on electronic correlation logic (as discussed in Section 2). CSMA leads to collision in *OOO* nodes, between locally inserted LFs and those from upstream nodes. Collided packets are recovered using the principle of splitting and storing a copy of a packet (prior to the collision) and then reinserting the same, when two such copies arrive at a node in overlapping time [1-3].

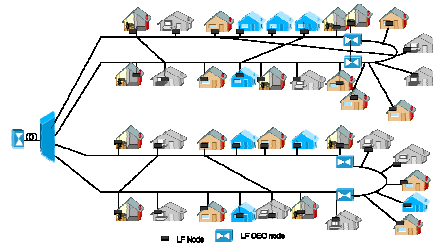


Fig. 2. Proposed Light-mesh network [3].

2. The Light-Mesh Test-Bed

The goal of the light-mesh test-bed is to demonstrate (1) optical packet transport; (2) recovery of Ether-packets (LFs) from collision; (3) service differentiation and conforming communication within delay bounds. The test-bed is a single node implementation with loop-back (as shown in Fig. 3-5) to facilitate full network behavior.

OMQ7.pdf

The line rate is assumed to be 1 GbE, (1.25 Gbps with 8b/10b encoding). LFs support standard Ethernet frames (<1500 bytes) and allow 802.3, 802.1P, 802.3ad and 802.3ah/ay implementations that support provider bridging using stacked VLANs. Two types of VLANs are assumed: Service VLAN (SVLAN) and Customer VLAN (CVLAN) which have unique *ids* (SID and CID).

Node architecture: The node architecture is shown in Fig. 4, and this node is assumed to be at a curb-site. The node is assumed to be on a string through which it is connected to the rest of the access network. The rest of the access network is emulated by an Ethernet packet generator with traffic type characterized by sub-wavelength granularity: 100~900 Mbps, intermittently bursty, and using VLAN tags. The generator is controlled by a TCL script through a master controller (Fig. 5). The node is connected to the string by two 80/20 passive couplers in 1x2 and 2x1 configuration. The first coupler, drop coupler (DC) is connected to a 1.25 GHz CDR circuit through a burst-mode receiver (BMRX) [5]. The second coupler called add coupler (AC) is connected to a burst-mode transmitter (BMTX) [5]. The BMTX and BMRX are connected to a Xilinx FPGA Virtex2pro through high-frequency microwave strip lines. Inside the FPGA are three hard-coded blocks: (1) bidirectional buffers, (2) counter and GEMAC, and (3) scheduler in addition to instantiation of a PowerPC.

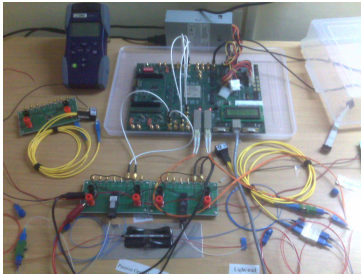


Fig. 3. Test-bed picture.

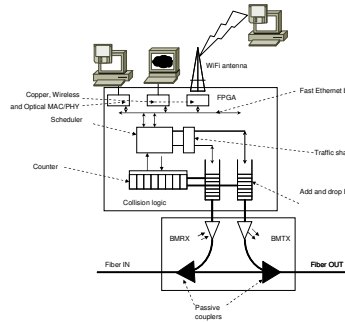


Fig. 4. Node architecture.

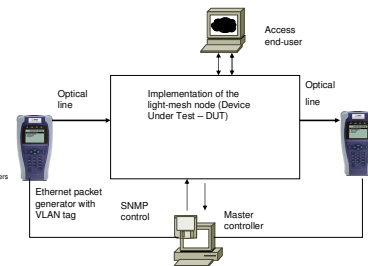


Fig. 5. Conceptual layout of the test-bed.

(1) *Bidirectional-buffers*: each buffer is 256 kb in size and stores Ethernet frames minus their headers. One of the two buffers is connected to the BMTX and the other one is connected to the BMRX. The buffers are controlled through the counter and GEMAC and the buffers send status information to the PowerPC.

(2) *Counter and GEMAC*: this is responsible for (a) communication and (b) collision recovery. For communication, we have TX and RX sub-blocks. The RX sub-block receives LFs from the BMRX. These LFs are processed by a PMA/PCS/GEMAC section in the sub-block. Each received LF triggers an increment in the counter. The PCS/PMA strips the LF into (1) the header, (2) VLAN tag(s) and (3) payload. The payload is sent to the buffer, while the header is sent to the scheduler. The VLAN tag(s) have information pertaining to the type of LF (data/control), the customer and the service (voice/video/data). The transmit/receive sub-block has a PCS section connected to a buffer through which it is connected to the BMTX/BMRX. Whenever an LF is transmitted/received, the counter is incremented. For each LF, the counter generates a start point and an end-point denoting when the LF starts and when it ends. If both the TX and RX counters are incremented in overlapping intervals (start and end points), then, it signifies that a collision happened. In such a case, a copy of the LF (which is locally stored in the buffer), is sent out for retransmission. If a received LF is of control type it implies either a scheduling request from other nodes or topology discovery, and this information is treated appropriately by a PowerPC within the FPGA.

(3) *Scheduler*: Since the light-mesh is a collision centric framework, the end-to-end delay depends on the total load in the system (and hence the total number of collisions). To enable delay sensitive services like voice and video to be provisioned we implement a QoS protocol that does efficient scheduling within the light-mesh. This distributed protocol (running at multiple nodes) enables scheduling of LFs into the network so as to reduce the number of collisions. Since the protocol is distributed, it is like a cooperative protocol whereby each node *self-regulates* its transmission with respect to every other node in the string. The scheduler gets a control packet (an LF with unique SID), that tells it of its number in the sequence queue (amongst the multiple nodes in the string). The node *sniffs* LFs (using drop-and-continue principle) from upstream nodes at the BMRX. When its assigned predecessor (in the sequence) has finished transmission it begins to transmit its own LFs. The scheduling of LFs in a string follows a round-robin pattern with necessary delay and bandwidth bounds as explained in Section 3.

End-user connection: In addition, the FPGA also drives customer premise equipment (CPEs) through a Fast Ethernet line and a WiFi access-point that provide bandwidth to end-users (Fig. 4). The WiFi AP is implemented through an off-the-shelf daughter-card that fits onto our designed board (as shown in Fig. 3).

PowerPC: an on-board power PC within the Xilinx FPGA (Virtex 2pro) is instantiated for providing intelligence required for control of the network (scheduling) as well as for service differentiation. The PowerPC assigns appropriate VLAN tags (CID/SID) to incoming and outgoing LFs. These tags enable service differentiation. LFs that are control packets and used to disseminate the network topology [1], are also processed

by the PowerPC, which builds and develops a logical topology. Neighbor discovery in a bus is non-trivial and is implemented as discussed in [1].

3. Providing Quality of Service to Flows/Packets in the Light-mesh

The nodes in the light-mesh network exchange control messages using in-band control by assigning control packets with a unique SID. These messages are instructive in providing QoS – particularly bounded delay and granular bandwidth to nodes in the network. It is important to provision services in the light-mesh by keeping collision within an acceptable upper bound. This protocol for QoS is based on our earlier time-slot scheduler called *delay sensitive smoothed round robin* (DS2R2) [4]. The principle of DS2R2 is as follows: time is divided into slots and a centralized arbiter is informed of the bandwidth requirement as well as the maximum delay tolerance of each *flow* (at every node). The centralized arbiter using the bandwidth and delay requests as inputs, computes two data-structures called the weight-matrix (proportional to the flow granularity) and the weight-spread sequence (essentially a Cantor set) which when cross-multiplied leads to an ordering assignment of time-slots in the network. However in the case of the light-mesh, time is not slotted and neither is a central arbiter available. Hence, each node characterizes the flow based on number of services (e.g. 4 voice lines each at 64 kbps, 2 video lines each at 6 Mbps), and sends the cumulative request to the end-node of the string (using control packets). Assuming N^2 connectivity, the end node communicates back to each node telling the node of (1) the order in which it is supposed to send data (preceding node information) and (2) the number of LFs (of a max size of 1500 bytes), it can send in each transmission. The traffic shaper (in Fig. 4), then transmits the appropriate number and size of LFs when the node is at the head-end of the transmit sequence.

4. Results

We performed experiments to showcase optical packet transport as well as provision services. We generate 3 types of traffic (voice, video and data) with maximum allowable latency of 25, 20 and 400 ms respectively. Our main goal was to measure delay with and without the QoS algorithm for different traffic mixes. To do so, we varied the mixes into three categories (with data, voice and video in proportions of 70:20:10; 60:30:10 and 50:30:20 respectively). Shown in Fig. 6 is the delay profile of the three traffic mixes as a function of network load. Load is computed as the ratio of bits sent into the network (by the node and the traffic generator) to the line-rate. In this figure, we have applied the QoS scheme and we observe that the (end-to-end) delay is less than 5 milliseconds even at high-loads. Intuitively the delay is more when there is a higher percentage of data-traffic. In Fig. 7, we take the traffic mix type 2 (60:30:20), and compare the delay results with and without QoS. Without QoS, the delay reaches an unacceptable high of 32 ms for unity load. The QoS technique on an average has 64 % betterment in lowering delay. Finally, we observe the utilization experienced in the network in Fig. 8. Shown in the figure are four curves for traffic mix type 2: (1) with QoS as observed in the test-bed, (2) with QoS as observed in theory (DS2R2), (3) without QoS and finally (4) without QoS but neglecting the collided packets. We observe that our theory and practical results are very close to each other (mean difference 8 %, max 15%) giving us confidence in this test-bed. In the case when we do not have QoS, the utilization hovers around 95%. This means that the entire bandwidth is being used for transport of both collided and normal LFs. If we neglect the collision we see the actual utilization in the network – as it degrades rapidly. This shows the effect of collision on utilization and the benefits of the QoS technique.

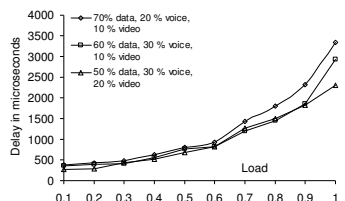


Fig. 6 Delay with multiple traffic mixes.

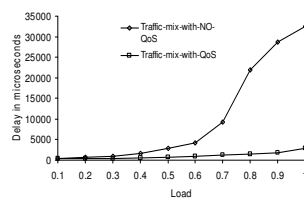


Fig. 7. Delay with and without QoS.

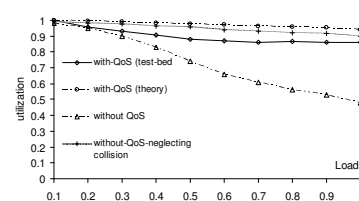


Fig. 8. Utilization with and without QoS

5. Conclusion

We summarize this paper as an implementation of optical packet transport using mature technology and achieving service differentiation, collision recovery and delay tolerance, particularly useful in the access area.

Acknowledgements: Authors acknowledge Profs Krithi Ramamritham, Vincent Chan and Bernard Menezes for their support and Xilinx/Avnet and JDSU for their generous equipment grant.

References

- [1] A. Gumaste et al, *Invited paper* Elsevier Optical Switching and Networking (OSN) Journal, March 2008
- [2] A. Gumaste and S. Zheng, *IEEE/OSA Journ. of Lightwave Tech.* Oct 2006. pp 82-96
- [3] A. Agrawal et al, *IEEE/OSA OFC 2008*, San Diego, CA TuF
- [4] P. Bafna et al, *IEEE/OSA OFC 2007*, Anaheim CA, TuGA
- [5] Y. Ota and R. Swarz, *IEEE/OSA Journ. of Lightwave Tech.* 1990, pp 57-73