On Control Plane for Service Provisioning in Light-trail WDM Optical Ring Networks

Ashwin Gumaste¹, Janak Chandarana¹, Paresh Bafna¹, Nasir Ghani² and Vishal Sharma³

¹Indian Institute of Technology, Bombay India.

²Tennessee Tech University, Cookeville TN USA

³Metanoia Inc, Mountain View CA. USA

Email: ashwing@ieee.org, janak@it.iitb.ac.in, paresh@it.iitb.ac.in, nghani@tntech.edu, v.sharma@ieee.org

Abstract: A light-trail is a generalized lightpath that enables multiple nodes to statistically share an optical communication path (wavelength bus). A light-trail is different from a lightpath on account of its unique node architecture – that enables formation of unidirectional wavelength buses. A node in a light-trail supports signal flow characteristics of drop-and-continue as well as passive addition leading to formation of the bus. Apart from the node architecture, another differentiation between light-trails and lightpaths is in the use of out-of-band control channel (Optical Supervisory Channel – OSC) for dynamic communication. The combined effect of a unique node architecture and out-of-band control channel enables light-trails to provide capability of sub-wavelength grooming, dynamic provisioning and optical multicasting. These features offered by light-trails are critical for next generation emerging applications such as VoIP, Video-on-demand (VoD), Triple-play and Pseudo-Wire Edge-to-Edge Emulation (PWE3). Current OSC based control channel requires engineering enhancement to provision dynamic and bandwidth efficient services over light-trails. We investigate into the control channel hierarchies in light-trail networks to provision these emerging services. Centralized, distributed, static and dynamic control mechanisms are proposed. Light-trail control for provisioning next generation services is considered through theory and simulation.

I. INTRODUCTION

Next-generation services such as triple-play, Video-on-Demand (VoD) and pseudo-wires are revenue intensive emerging applications for future optical networking [12]. Such services require a combination of features that include dynamic provisioning, optical multicasting and reduced capital and operational costs at the optical layer. To provision these services, the optical layer must be able to provide a paradox of requirements – of being able to dynamically facilitate bandwidth (on-demand) while concurrently maximizing the network utilization (cost reduction). While the former calls for sufficiently advance networking technologies and control schemes, the latter requires sub-wavelength grooming (preferably for cost purposes in the optical domain) – requiring a paradigm shift from the conventional lightpath communication. Light-trails [1, 2] a generalization of a lightpath have been proposed to provide sub-wavelength grooming [3], dynamic provisioning [3, 4], and a low-cost solution [1, 2]. Due to its unique architecture, a light-trail is analogous to an optical wavelength bus – nodes within a light-trail arc, are able to communicate with one-another without any optical-switching. To support the wavelength bus, nodes have properties of allowing an incoming signal on a

wavelength to be dropped-and-continued, as well as to add signal passively, when upstream nodes are not transmitting data. The resulting wavelength bus can cause potential conflicts amongst transmitting nodes. To avoid conflicts and for management of light-trails we use an out-of-band control channel. The control channel is optically dropped and electronically processed at every node and then reinserted into the optical network.

While optical bus technology has been proposed and investigated [8], what makes the case for light-trail unique is the existence of an out-of-band control channel enabling arbitration and service provisioning within the bus. The control channel enables nodes to statistically share the light-trail wavelength bus. The control channel in a light-trail is necessary not only for arbitration and for achieving efficient performance but also to enable provisioning of services. Meeting service requirements like latency, guaranteed-throughput, jitter and pseudo-wire emulation using the control channel as a provisioning and signaling mechanism, presents an interesting engineering problem in light-trail networks and is the focus of this paper.

Organization of this paper: In section II, we describe the light-trail node architecture as well as the control plane design issues. Section III introduces approaches for realizing the control plane. Section IV practically applies control schemes proposed to include service provisioning as well as showcases simulation results while Section V concludes the paper.

II. LIGHT-TRAIL NODE ARCHITECTURE AND CONTROL

A light-trail is analogous to a shared wavelength bus, and enables multi-point-to-multi-point communication at the optical layer supported through an out-of-band control channel. Shown in Fig. 1 is an implementation of a light-trail node for optical ring networks. A comparative reference model of lightpaths and light-trails is shown in Fig. 2.

The node architecture in Fig. 1 is that of a Reconfigurable Optical Add Drop Multiplexer (ROADM) converted into a light-trail capable node. Composite WDM signal enters the node and is de-multiplexed into its constituent wavelengths. Each wavelength feeds to a Light-trail Optical Retrieval Section (LORS) that comprises of two passive couplers separated by an optical switch. LORS enables a node to access data in the light-trail. The first coupler is called the drop-coupler (used to passively drop optical signal), while the second coupler is called the add-coupler (used to passively add optical signal). The first coupler is in a 1x2

configuration while the second coupler is in a 2x1 configuration. The function of the first coupler is to drop-and-continue the incoming optical signal (light-trail), while the second coupler enables data to be locally inserted into the light-trail.

Each LORS is connected to a transponder that enables nodes to send and receive data into and from the time-shared light-trail. To implement time-sharing of the wavelength bus each transponder is assumed to exhibit burst-mode operation as well as electronic queueing, and in [7], we have demonstrated such a light-trail transponder calling it a *burstponder*.

The unidirectional light-trail (bus) is setup by blocking the wavelength between the two extreme nodes called the convener node and the end-node (as shown in Fig. 2). The process of setting up/tearing down of light-trails requires optical switch configuration and hence is infrequent on account of the time lost due to (stable but slow) optical (mechanical) switching technology. Once a light-trail is setup, nodes communicate with one-another under the condition that no two nodes transmit data in overlapping time intervals. When a node transmits data into the light-trail it forms a *connection*. Connection provisioning i.e. setting up and tearing down is conducted by the out-of-band control channel, and does not require optical switching. Connection provisioning is dynamic due to its switch-less mechanism. We term creation/destruction of light-trails as hard-provisioning and connection provisioning as soft-provisioning. Light-trail and connection provisioning have been theoretically studied in [6].

Fig. 1. Light-trail node architecture and control processing

In [1], is proposed a Just-Enough-Time [19] based control mechanism that allows nodes to use light-trail bandwidth based on their position with respect to the convener

of the light-trail. This approach gave a higher probability of success to nodes that were closer to the convener than to nodes closer to the end-node, leading to issues of fairness and hence inability to provision services [1]. Solution to some of the fairness and dynamic bandwidth allocation problems were solved in [4] by proposing a ‘*dual auction*’ mechanism for bandwidth allocation in light-trails. Duality was exhibited as:

- Setting up/tearing down connections (soft provisioning)
- Setting up/tearing down light-trails (hard provisioning)

The dual aspect was shown to yield fairness as well as stability [4]. The light-trail system results in a bifurcation of the control plane from the data plane. The control plane is housed in a single channel (typically a non-ITU grid wavelength – 1510 nm) and the remaining wavelengths (ITU grid) are relegated to the data plane. The control plane due to OEO (at every node), is synchronized with respect to all nodes. This control plane initiated synchronization allows us to assume a light synchronization (with large guard-bands) even at the data plane. This allows the data plane to be time-slotted [11, 13]. Time-slotting means that burstponders can begin and end transmission at the boundaries of predefined slots since their operation is controlled through the control channel. With this understanding, we are in a position to discuss protocols that can be housed in the control channel for bandwidth arbitration, light-trail growth and service provisioning.

Fig. 2. Comparison of Lightpaths and Light-trails

Fig. 3. Control channel hierarchy in light-trails

Fig. 4. Example of SRR functioning with WM and WSS from SRR [16]

Shown in Fig. 3 is a hierarchy of control channel implementations that we investigate in this paper. Protocols for out-of-band arbitration are primarily of two types: centralized and distributed. Centralized protocols require a

predetermined node in the light-trail to act as an arbiter. Distributed protocols do not have the arbiter requirement but make use of a distributed agreement mechanism for bandwidth arbitration. Centralized protocols can be static (periodic assignment) or dynamic (on-demand assignment).

III. CONTROL TECHNIQUES FOR LIGHT-TRAIL NETWORKS

In this section we consider different control techniques for service provisioning in light-trail networks.

III. A. Centralized Control Schemes

In the centralized model, a pre-selected node acts as a bandwidth arbiter for each light-trail. Nodes send their request to the arbiter, and are informed about provisioning their connections either in advance (static) or in real-time (dynamic).

III. A. 1. Static Centralized Control Scheme (SCCS)

In this scheme, nodes within a light-trail send their bandwidth request in form of service needs to an arbitrating node. Each new arriving request is recorded in an *assignment* database at the bandwidth arbiter. The arbiter, through the control channel notifies each node of the transmit time and duration for which the node should initiate a connection. The system continues in the same state until a node admits a new service or changes its bandwidth needs. New requests are incorporated by reconfiguration of the assignment database. If however, a node has a bandwidth request that cannot be incorporated, the arbiter signals to the proposing node to set up or look for a new light-trail. The scheme is static: mapping of service requests to bandwidth assignment is done just once – during admission of the service at the ingress node. Once the service is admitted, no further requests are sent till the service is terminated at the node or the light-trail is torn down.

The light-trail data-plane is time-slotted, implying that we are required to find a connection ordering (scheduling) sequence for all the requests that the arbiter receives. We make use of the *Smoothed Round Robin* (SRR) [16] scheduling protocol that sequences flows, based on their weights, in a way to guarantee short-term fairness as well as reduce the effects of burstiness in data traffic, with $O(1)$ complexity. SRR uses two data structures called *Weight Matrix* (WM) and *Weight Spread Sequence* (WSS) to enable scheduling of flows. The WM is a binary representation of flows normalized according to their weights. The WSS is a sequence that resembles the interleaved elements in each level of a Cantor set, when viewing the set from left to right (see Fig. 4 for details). While SRR provides simplicity, short-term fairness and smoothes effects of burstiness, it however fails to provide scheduling within delay based deadlines (critical for provisioning services). We proposed *Delay Sensitive Smoothed Round Robin* (DS2R2) [14] – a $O(1)$ scheduling algorithm that not only considers the weight of the flows (to be scheduled) but also takes into consideration deadlines for each different flow. The DS2R2 scheme facilitates provisioning of revenue bearing services over networks that

have low connection setup time. Naturally, DS2R2 is ideal for bus networks with centralized arbitration since within a bus, connection provisioning involves no switching resulting in maximum efficiency. The following notation is used for working of DS2R2:

λ_{ik}^h , Data rate for h^{th} service at the i^{th} node for transmission in the k^{th} light-trail with capacity $C(k)$.

Δ_h , Maximum allowable delay for h^{th} service.

Preprocessing for DS2R2

$T_s = \frac{\lambda_{0k}^h}{C(k)}$, Where $\lambda_{0k}^h = \min_{\forall h, \forall i \in k} (\lambda_{ik}^h)$, and T_s is the time-slot

given for transmission to every node in the light-trail. The dependency on λ_{0k}^h enables T_s and hence cycle time to conform to delay latency of all the services.

$Q_{ik}^h = \frac{\lambda_{ik}^h \Delta_h}{C(k)T_s}$, Number of time-slots of duration T_s , required in

the interval Δ_h by h^{th} service of the i^{th} source in the k^{th} light-trail to ensure that both delay and bandwidth constraints for the flow are satisfied.

$Q_{ik}' = \Delta_{\min} \frac{Q_{ik}^h}{\Delta_h}$, Number of time-slots of duration T_s , required

in interval $\Delta_{\min} = \min_{\forall h} (\Delta_h)$ by h^{th} service of the i^{th} source in the k^{th} light-trail to conform to the delay bound of Δ_h for every packet (of the h^{th} service).

Explanation of preprocessing for DS2R2: Parameter Q_{ik}' is computed on the following principle: if the flow of the h^{th} service at the i^{th} node in the k^{th} light-trail is assigned Q_{ik}' time-slots of duration T_s in interval $\Delta_{\min}$, then both bandwidth and delay requirements of that flow will be satisfied. This results from deriving Q_{ik}' from Q_{ik}^h , which is computed taking into consideration both delay and bandwidth constraints of every flow. Q_{ik}' differs from Q_{ik}^h , since Q_{ik}^h assumes a cycle time of Δ_h , while Q_{ik}' assumes a normalized cycle time of $\Delta_{\min}$ (refer [14] for more details). Thus, from Q_{ik}' we can construct WM:

$$WM[i] = w_i = Q_{ik}', \forall i \in k$$

Using WM and WSS (as shown in Fig. 4) we can schedule the flows in a light-trail sufficing the delay and bandwidth requirements of each flow. We further propose two modifications that enable greater fairness and admission control.

Static Centralized Adjusted Latency Scheme (SCALS): In this scheme, when a node sends its bandwidth request to the arbiter, the arbiter updates its assignment database with this new request such that, it adjusts the previously made

assignments in a way to enable the latency requirement of this new request to be incorporated (while preserving previous requests). The arbiter computes a sequence of connection ordering and number of slots allocated, such that, the ordering satisfies the latency request and the number of slots satisfies the bandwidth (flow) requested by all the nodes (based on the DS2R2 algorithm). The ordering is such that the duration between two successive connection assignments for a node is less than the latency requested by the service provisioned at the node [14], so that service time-out does not occur. Shown next is pseudo code for SCALS. Assume that a flow $\lambda_{ek}^{h_e}$ at node e is required to be incorporated into light-trail k . The procedure is as follows:

```

SCALS (adding flow  $\lambda_{ek}^{h_e}$  to light-trail  $k$  and
obtaining connection schedule)
if ( $\Delta_{h_e} < \Delta_{\min}$ )
     $\Delta_{\min} = \Delta_{h_e}$  do preprocessing (re-compute
 $T_S$ ,  $Q_{ik}^h \forall i \in k$  and  $Q_{ik}^{h_e} \forall i \in k$  values, generate WSS)
else
    compute  $Q_{ek}^{h_e}$  and  $Q_{ek}^{h_e'}$  values
    add row of  $Q_{ek}^{h_e'}$  to  $WM = WM \cup \text{Bin}(Q_{ek}^{h_e'})$ ,
where  $\text{Bin}(Q_{ek}^{h_e'})$  represents binary representation
[14] of  $Q_{ek}^{h_e'}$  and  $U$  represents addition of a row
to the existing WM as well as zero-bit
stuffing (if necessary) to make the WM
regular.

```

Explanation: The first *if* statement states that, if delay value of the new flow is lesser than minimum delay value of all existing flows, then we are required to do pre-processing for all services and nodes. The modified values as well as generating schedule are obtained thus incorporating the new flow in addition to existing flows. Otherwise, if the permissible delay value is greater than the minimum value then we have to compute the associated Q , Q' values just for the new flow and update the WM from which point DS2R2 does the scheduling of flows.

Static Centralized Adjusted Bandwidth Sharing (SCABS): In this scheme, when a node sends its bandwidth request to the arbiter, the arbiter updates its assignment database with this new request such that, it adjusts previously made assignments in a way to enable the bandwidth request to be serviced in a *fair* (equally likely) manner. The idea is to allocate the light-trail bandwidth in a way that the ratio λ_i^r / λ_i is the same for all nodes i ; where λ_i is the requested bandwidth by each node i in the light-trail and λ_i^r is the bandwidth allocated by the arbiter in response to this request. Shown next is the pseudo code for the functioning of SCABS:

SCABS (adding flow $\lambda_{ek}^{h_e}$ to light-trail k and obtaining connection schedule)

```

Let  $\alpha = \left( \lambda_{ek}^{h_e} + \sum_{i \in k, \forall h} \lambda_{ik}^h \right) / C(k)$ 
if  $\alpha > 1$ 
    compute  $\lambda_{\text{mod}}[ik]^h = \lambda_{ik}^h / \alpha$ ,
    preprocess (re-compute  $T_S$ ,  $Q_{ik}^h$  and
 $Q_{ik}^{h_e} \forall i \in k$  values, generate WSS)
elseif  $\alpha < 1$ 
    compute  $Q_{ek}^{h_e}$  and  $Q_{ek}^{h_e'}$  values,
    add row of  $Q_{ek}^{h_e'}$  to WM and obtain
 $WM = WM \cup \text{Bin}(Q_{ek}^{h_e'})$ , where  $\text{Bin}(Q_{ek}^{h_e'})$  represents
binary representation of  $Q_{ek}^{h_e'}$  number of columns
(if required).

```

Explanation: The *if* statement signifies whether the total flow (including the newly added flow $\lambda_{ek}^{h_e}$) is greater than the capacity of the light-trail, then we scale down (*mod*) each flow from λ_{ik}^h to $\lambda_{\text{mod}}[ik]^h$ and then perform preprocessing. *Else*, if the total flow including the added flow is less than the capacity of the light-trail, then we perform preprocessing only for the added flow $\lambda_{ek}^{h_e}$.

III. A. II. Dynamic Centralized Control Scheme (DCCS)

In the dynamic centralized scheme the arbiter for every light-trail arbitrates on a real-time basis. Nodes send their bandwidth requests ahead in time to the arbiter. The arbiter decides which node should be given transmission rights in the light-trail and signals this assignment back to all the nodes. The winning node then provisions its connection. The connection could either occupy a time-slot or could be provisioned until another node's request has a higher utility than that of this node (in an un-slotted system).

The critical aspect in dynamic centralized control scheme is in the computation of node bandwidth and service request into numerical values to facilitate arbitration. We call these numerical values *valuations*. The *valuation* that a node sends to the arbiter must reflect both the bandwidth requirement (quantity), as well as service requirement (quality).

Computing Absolute and Service Valuation: We now show how to compute absolute and service valuation. Consider a $n(k)$ node light-trail, and let each node have the architecture as shown in Fig. 1 with bi-directional control (2-fiber ring). Assume a synchronized and slotted control channel. Every burstponder, part of the light-trail node has an electronic buffer to queue packets (representing a service).

Let Buff_m be the maximum size of the buffer. Let $d_1, d_2, \dots, d_S$, represent the maximum allowable delays of the S possible services. Let t_{sp} and t_{hp} represent the times required for soft-provisioning and hard-provisioning. In [7, 9] the

values of t_{sp} and t_{hp} are $10 \mu s$ and $3 ms$ respectively for a 1-10 Gb/s light-trail.

Our desire is to compute the absolute and service valuation that a node i sends either to an arbiter (in the centralized case) or to other nodes (in the distributed case as explained subsequently) for connection provisioning in light-trail k .

Absolute valuation has two components – a quantitative bandwidth requirement component, and a qualitative services requirement component. The absolute valuation is the maxima of both these components when they are individually normalized as shown next.

Normalized Quantitative Component: defined as the ratio of size of buffer at a time t to the maximum size ($Buff_m$).

$$Quant_{ik}(t) = \frac{Buff_{ik}(t)}{Buff_m}$$

where $Buff_{ik}(t)$ represents the buffer size at node i for transmission in light-trail k at time t . Note, $Quant_{ik}(t)$ is bounded between 0 and 1. Note that $Quant_{ik}(t)$ follows a Poisson distribution (assuming arrival process of packets into the burstponder to be Poisson).

Normalized Qualitative Component: defined as the normalized waiting time before which a node must be serviced so that the service (delay) guarantees are met. The following explanation clarifies the above definition:

Let $q_{ik}(t)$ be the time elapsed since the first packet (belonging to any of the S services) entered the buffer $Buff_{ik}$.

At any given time, let us denote $Z_{ik}(h)$ as the time elapsed since the first packet of service type h entered buffer $Buff_{ik}$. Then the value, $d_h - Z_{ik}(h)$ denotes the time that packets corresponding to service h must be transmitted in the light-trail else they will be timed-out. Hence the value, $\min_{\forall h}[d_h - Z_{ik}(h)]$ computed over all h corresponds to a critical limit [3], within which the node must be allocated a connection to meet the service requirement. To ensure fairness, this critical limit is normalized with respect to the critical limits of all the nodes in the light-trail, or simply over the period of time it gets its connection as compared to the connections received by other nodes in the light-trail. Hence, we compute normalized qualitative component as:

$$Qual_{ik}(t) = \frac{q_{ik}(t)}{q_{ik}(t) - \min_{\forall h}[d_h - Z_{ik}(h)]}$$

Note, $Qual_{ik}(t)$ is bound between 0 and 1 as the second term in the denominator as well as the numerator cannot be negative. Further, note $Qual_{ik}(t)$ has a sigmoidal-like distribution [15].

Therefore, absolute valuation is denoted by:

$$Abs - Val_{ik}(t) = \max[Qual_{ik}(t), Quant_{ik}(t)]$$

The service valuation is due to the normalized qualitative component only:

$$Serv - Val_{ik}(t) = Qual_{ik}(t)$$

III. B. Distributed Control Scheme

In the distributed schemes, there is no central arbiter. Further, the distributed scheme is dynamic. An example of the distributed scheme is the $(n-2)$ Distributed Control Scheme ($n-2$ DCS) which is now explained:

In the $n-2$ DCS, we replicate the dynamic centralized scheme of nodes converting their requests into valuations – with a twist. Instead of an arbiter deciding which node would set up a connection, the nodes themselves arbitrate through the following method: each node sends its valuation to all other nodes in the light-trail at predefined intervals through the OOB control channel.

For an n -node light-trail, a node receives $n-2$ valuations (from all the nodes in the light-trail except itself and the end-node – since the end-node does not transmit data). Upon receiving $n-2$ valuations the node infers that it has the rights to form a connection (say in the next time-slot for a time-slotted system), if, all the valuations that it receives, are less than its own valuation. The node then sends a *proclamation message* through the control channel to all the nodes informing of its success and begins transmission in the next time-slot.

The advantage of this system over the previous one is that of reliability – not requiring an arbiter. The disadvantage of the distributed system is the time lost in convergence. Thus, time to provision connections becomes critical; especially when we consider that the electronic buffers in the burstponders are finite.

The $(n-2)$ DCS requires that the nodes exchange their valuations periodically. The periodicity of this exchange is predefined and fixed for both slotted and un-slotted systems. For 1 Gb/s light-trail with 3 millisecond time-slots, we have found [15] that 99.999 % of requests converge within the 3 ms duration for a light-trail of size < 15 nodes (metro networks). Similar results have been seen for 10 Gb/s light-trails with 2 millisecond time-slots. Another assumption of the $n-2$ DCS is that the arbitration is a self evaluation process – a node has to fairly report if the $n-2$ valuations it received were lesser than its own valuation.

IV. THE LIGHT-TRAIL CONTROL PLANE AND SERVICE PROVISIONING: SIMULATION AND RESULTS

Our emphasis in this section is on the adaptation of the light-trail control plane to enable provisioning of services as well as showcase numerical results through simulation. We consider the following four services: VoD, PWE3, VoIP and Triple Play (combination of voice, video and data). We show the protocol stack for provisioning services over light-trails in Fig.5. We further develop a simulation model for numerical results that provision services over light-trails using the control protocols mentioned in Section III. Our objective is to measure performance of different services over a light-trail ring network using control schemes suggested earlier. In the simulation model we consider a 16-node (metro) 2-fiber ring network with 32 wavelengths (in each fiber) and 1 wavelength in each fiber relegated for control (at 155 Mbps rate – ITU

OSC standard). Data-rate is assumed to be 1 Gb/s. Time for setting up a light-trail is 4 ms (inclusive of 2 ms optical-mechanical switch configuration time), while that for setting up a connection is 10 microsecond [9, 11]. Time-slots (assuming a slotted system) at the data plane are adjustable between 2-5ms duration. Buffer size in *burstponder* is 8Mb.

Traffic generation: we use three types of packet generators at every node: a voice generator based on superimposed Poisson sources, a data generator based on superimposed Pareto sources and a video generator superimposed by 70 % Pareto and 30 % Poisson sources. The maximum allowable latencies are: 50 ms for voice (VoIP), 75 ms (video) and 400-800 ms (data), depending on the simulation model discussed later. Different control schemes are simulated as well as different services are provisioned. Latency, efficiency and jitter are measured.

Fig. 5. Protocol stack over light-trails: Data traffic, pseudo-wires and VoD.

Fig. 6. Video-on-Demand over light-trail networks

IV. A. Video-on-Demand (VoD) model

The architecture for VoD is shown in Fig. 6. Nodes in the light-trail metro ring network are assumed to be either a *Video Hub Office* (VHO) or a *Video Serving Office* (VSO). When a customer requests for a video file, it is assumed that the customer is connected to a VSO through an access/first-mile network. The VSO sends the request through a metro network (a light-trail WDM network) to the VHO [12]. The VHO sends the video file over the metro network to the VSO; which then streams it to the customer.

Since the light-trail network is statistically shared, the video connections have to be given priority over data traffic; also video connections are on an on-demand basis, signifying the need for dynamic provisioning. The access-time for a video-file (time between a request being issued by a customer, and the VSO streaming the file) is of critical importance [17]. Through simulation, we have observed that access time (defined as the time between a request being sent and beginning of file reception) primarily a function of services present in the network as well as load. A network, in which

voice traffic is more dominant than video traffic leads to higher access times than for networks in which data traffic is more dominant.

The protocol stack for VoD is shown in Fig. 5. We show in Fig.7 the jitter experienced for a video file as a function of network load for file sizes between 400 MB and 750 MB (typical movie). As can be seen from Fig.7, jitter values obtained are less than 30 ms – the standard [18] acceptable jitter norm. The measurements for jitter assumed a 30 % video, 10 % voice and 60 % data traffic. One would expect that with large percentage of data traffic in the network, jitter for video traffic would suffer, but that is not the case in light-trail networks with the enhanced control plane. The reason for jitter control is that the protocol takes into consideration qualitative and quantitative aspects of the service.

Also we can observe from Fig.7 that dynamic control schemes have poor jitter values (though they lead to more efficient channel usage) than static schemes (SCALS and SCABS). Amongst the static control schemes SCALS outperforms SCABS in terms of jitter values – as it takes into account latency and hence jitter of the service being provisioned. In fact, at high network load SCABS has similar result to DCCS though the efficiency of SCABS (as will be shown later) is lesser than DCCS. This presents a design choice between SCALS and DCCS for static and dynamic control schemes.

We also show in Fig. 8 the access time in networks with different traffic mixes. Despite statistical sharing, we observe that the access time is lesser than the 10 ms for loads up to 50 % and is about 80 ms for a heavily loaded network which is within conformity of access time for video files [18].

Fig. 7. Jitter for video-on-demand files (for all 4 control schemes)

Fig. 8. Access time for video traffic as a function of load

IV. B. Pseudo-wire Edge-to-Edge Emulation (PWE3) model

PWE3 is used for transport of TDM and voice-centric services (T1/T3) over a packet network. IETF RFC 3985

(pseudo-wire architecture) defines the packet network as either an IP network or an IP/MPLS network – in which the optical layer is assumed to provide for *circuits* between IP or MPLS routers. With the introduction of light-trails, we require a further enhancement since the optical layer by itself is statistically shared and hence leads to a queueing penalty (with a trade-off of optical grooming, dynamic provisioning and low-cost). Despite the queueing penalty the improvement in efficiency is a strong motivation to investigate PWE3 over light-trails.

Shown in Fig.9 is the protocol stack of PWE3 over light-trails. Note that at the optical layer, the PWE3 control information is passed through the light-trail OOB control channel while the PWE3 data is passed through light-trails (statistically shared amongst multiple nodes). The generic pseudo-wire architecture [10] assumes customers to connect to a *Service Provider* (SP) through their *Customer Edge* (CE) which is typically a layer-2 switch. The SP creates an *Attachment Circuit* (AC) between the CE and its own metro networking infrastructure represented by the *Provider Edge* (PE) (as shown in Fig. 9). The pseudo-wire is emulated between two PEs.

In the generic case, bandwidth in the WAN or the PSN (Packet Switched Network) is pre-allocated – guaranteed at the optical layer (typically SONET tunnels). However, in the case of a light-trail based network, the bandwidth between PEs is not pre-allocated but is dynamically shared with other services as well as with other nodes. This means at the PE, the data on the incoming pseudo-wire is queued in the burstponder until a connection is provisioned. To do so, the incoming AC at the ingress node is directed into a burstponder for queueing. The burstponder requires modification to extract the pseudo-wire control word [10]. Upon extraction of the control word, the burstponder computes a valuation that reflects the latency, jitter and bandwidth requirements to provision the pseudo wire. Our simulation results show pseudo-wire performance over light-trails by using dynamic control protocols (DCCS) and compare the performance with IP router based networks in Fig. 10. For the IP-router network we measure performance over an OPNET modeler.

For simulation we assume the control scheme to be DCCS. While there is some improvement in terms of latency using IP routers, there is a significant *over-provisioning* penalty incurred. Over-provisioning is necessary to maintain deterministic TDM performance over a packet/statically multiplexed network. This signifies the wastage of bandwidth in the IP network for PWE3. Light-trail due tight coupling between control and data results in bandwidth saving. Shown in Fig. 10 is a comparison of light-trails and IP networks in terms of efficiency of over-provisioning as a function of load, as well as the delay incurred in provisioning 20 % PWE3 traffic. As can be seen, IP networks require more over-provisioning to facilitate PWE3 traffic while light-trails due to tight data-control coupling require lesser over-provisioning. The delay incurred in provisioning PWE3 using light-trails is greater than IP networks due to queueing penalty. However,

the penalty is still lesser than the acceptable PWE3 norm of 90 ms [10].

IV. C. Triple Play Services over Light-trail Networks

The triple play of voice, video and data implies a trade-off in bandwidth needs. Voice (in circuits or in packets) represents a finely granular service with stringent latency requirements, data represents voluminous bandwidth needs with less stringent latency; while video traffic can potentially be both voluminous and latency sensitive. Transforming these requests through a normalization process that enables statistical time-sharing of the light-trail bandwidth is a challenge. The method of valuations suits our purpose of provisioning triple play services by using an efficient control scheme.

Amongst the centralized and distributed control mechanism, it is the centralized mechanisms that are apt for triple play services – the reason being their ability to provision the network (in terms of bandwidth provisioning) to the variable traffic (both quantitative and qualitative) demands. Our simulation results delve into triple play provisioning using the centralized scheme. Shown in Fig. 11 is triple play performance (efficiency) as a function of network load for two different traffic mixes.

Fig. 9. PWE3 over light-trails

Fig. 10. PWE3 comparison over light-trails and IP routers for delay and over-provisioned bandwidth

Fig. 11. Triple play performance over light-trail network

IV. D. Comparison of Control Schemes in Light-trail Networks

In this sub-section, we measure the performance of different control schemes from the perspective of efficiency and latency. Efficiency is measured as the average amount of time the channel carries useful information. We observe (in Fig. 12) that the static schemes have better efficiency than the dynamic schemes – which is quite intuitive. However, we also observe that the centralized dynamic scheme (DCCS) is almost as efficient as SCABS because of real-time arbitration and efficient computation of valuation. In contrast, for average experienced latency (shown in Fig. 13) the dynamic schemes out-perform the static schemes as connections are provisioned on a need-to-send basis. For low-loads all the control schemes (both static and dynamic, centralized and distributed) have similar performance. However, as load increases the static schemes incur a greater latency penalty due to their deterministic nature in provisioning connections.

Fig. 12 Efficiency of different control schemes with 30 % voice, 50 % data, 10 % video and 10 % PWE3 traffic

Fig. 13. Latency experienced through different control schemes in light-trail networks

V. CONCLUSION

The recently proposed light-trail technology provides an alternative optical layer solution that facilitates dynamic provisioning, sub-wavelength grooming and an evolutionary low-cost platform – the key enablers for emerging services. The key differentiation in light-trail technology is in the use of the OOB control channel (along with node architecture) that enables successful dynamic bandwidth provisioning. In this paper, we propose a wide range of control protocols that ensure efficient bandwidth provisioning in light-trails. The protocols proposed are bifurcated into centralized and distributed types based on their operation. Centralized protocols are further bifurcated into static and dynamic types. We adapt our recently proposed DS2R2 scheduling algorithm to static centralized protocols. We also adapt a valuation based mechanism for dynamic centralized as well as distributed

protocols. Performance metrics and protocol stacks for provisioning services are also discussed. Services such as PWE3, VoD, triple play are considered. The paper summarizes the importance of control protocols and provides good design choices for successful implementation of light-trail WDM optical ring networks.

Acknowledgement: Authors thank Prof. Krithi Ramamritham at IIT Bombay for his encouragement and support.

REFERENCES

- [1] A. Gumaste and I. Chlamtac, "Light-trails: An Optical Solution for IP Transport," *OSA Journal on Optical Networking* May 2004, 864-891.
- [2] A. Gumaste and I. Chlamtac, "Light-trails: A Novel Conceptual Framework for Conducting Optical Communications," *IEEE High. Perf. Swit. And Rout.HPSR 2003* Torino Italy June 2003.
- [3] A. Gumaste and S. Zheng, "Next Generation Optical Storage Area Networks: The Light-trails Approach," *IEEE Communications Magazine* Mar. 2005 Vol. 21. No. 3. pp. 72-79 and references therein.
- [4] A. Gumaste and S. Zheng, "Dual Auction (and Recourse) Opportunistic Protocol for Light-trail Network Design," *IEEE Wireless and Opt. Commun. Conf.(WOCN)* Bangalore, India 2006.
- [5] J. Stankovic, M. Spuri, K. Ramamritham and G. C. Buttazzo, *Deadline Scheduling for Real-time systems: EDF and Related Algorithms*, Kluwer Academic Publishers, 1998.
- [6] A. Gumaste, G. Kuper and I. Chlamtac, "Optimizing Light-trail Assignment to WDM Mesh Networks," *IEEE Proc. of LANMAN*, San Francisco 2004.
- [7] P. Palacharla, A. Gumaste, T. Naito, "Implementation of Burstponder Card for Ethernet Grooming in Light-trail WDM Networks," *IEEE Proc. of Intl Conf. on Commun. ICC 2006* Istanbul.
- [8] B. Mukherjee and J. Meditch, "The p_i Persistent Protocol for Unidirectional Broadcast Bus Networks," *IEEE Trans on Commun. Vol. 36* No. 12 Dec. 1988.
- [9] A. Gumaste, P. Palacharla and T. Naito, "Performance and Demonstration of Light-trails in Shared Wavelength Optical Networks (SWONs)," *IEE Proc of Euro. Conf. on Opt. Commun. ECOC 2005*.
- [10] S. Bryant et. al, "Pseudo wire Edge-to-Edge Emulation (PWE3) Architecture," *IETF RFC 3985*.
- [11] A. Gumaste, P. Palacharla and T. Naito, "Experimental Demonstration for Static and Dynamic Control for Light-trail Networks," *11th Opto-Electronic and Communications Conference (OECC)* Taiwan 2006.
- [12] *SBC Lightspeed Project*: (online source) <http://att.sbc.com/gen/press-room?pid=5838>
- [13] A. Gumaste, J. Chandarana N. Ghani and V. Sharma, "Service Aware Control Schemes for Light-trail WDM Networks," *KReSIT IIT Bombay Tech Report 2006*.
- [14] P. Bafna, A. Gumaste and N. Ghani, "Delay Sensitive Smoothed Round Robin Scheduler (DS2R2) for High-Speed Optical Networks," *to appear IEEE Communication Letters*
- [15] A. Gumaste et al, "Two Stage Auction Algorithm for Fair Bandwidth Allocation and Topology Growth in Light-trail based WDM Networks," submission to *IEEE Journ of Lightwave Tech.*
- [16] G. Chuanxiong, "SRR: An $O(1)$ time complexity packet scheduler for flows in multi-service packet network" *ACM SIGCOMM*, Oct 2001.
- [17] J.-F. Paris, "A Simple Low-Bandwidth Broadcasting Protocol for Video-on-Demand," In *Proc. 8th Intl Conference on Computer Communications and Networks (IC3N'99)*, Boston-Natick, MA, Oct 1999
- [18] Online reference: *Optimizing Video Transport in Your IP Triple Play Network*: Cisco Systems Inc. http://www.cisco.com/en/US/products/hw/routers/ps368/products_white_p aper0900aecd80478c12.shtml
- [19] C. Qiao et al, "Labeled Optical Burst Switching LOBS: IP over WDM Networks," *IEEE Commun. Magazine*. Vol. 18 Sept 2001.