Burst Contention Avoidance Schemes in Hybrid GMPLS-enabled OBS/OCS Optical Networks

Jordi Perelló, Salvatore Spadaro, Jaume Comellas and Gabriel Junyent
Optical Communications Group (GCO), Signal Theory and Communications Department (TSC)
Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, 08034 Barcelona (Spain)
Tel: (+34) 93 401 7179, Fax: (+34) 93 401 7200
Email: {jperello, spadaro, comellas, junyent}@tsc.upc.edu

***Abstract*—Hybrid optical network architectures, combining benefits of optical circuit and burst switching technologies, become a natural evolution to improve overall network performance while reducing related costs. This paper concentrates on preventive contention avoidance schemes to decrease burst loss probability at the OBS layer of such hybrid network scenarios. Into operation, the proposed solution locally reacts to highly loaded downstream node situations by preventively deflecting bursts through a less loaded neighbor. Two different approaches for disseminating adjacent nodes state information are presented and extensively evaluated. In the first approach, current node state information is propagated downstream in the burst control packet, keeping pace with OBS traffic dynamics. The second approach targets at lower control overhead. In this case, averaged node state statistics are included in the Hello messages of the GMPLS Link Management Protocol (LMP) protocol, which are exchanged between neighboring nodes over the OCS control layer every 150 ms. The obtained results validate the applicability of both approaches. Moreover, they indicate that, depending on the mean burst size, either one or the other approach is favorable.**

I. Introduction

The proliferation of next-generation broadband data applications like video conference, HDTV or telemedicine, is moving network design from the traditional layered approach to multi-service network architectures. Therein, the IP protocol plays a unifying role, seamlessly integrating each different service onto the same transport network infrastructure. To make the transmission of such generated huge amounts of information possible, different all-optical transport network architectures have been proposed, opened up by advances in Wavelength Division Multiplexing (WDM) technologies.

Next-generation Optical Circuit Switching (OCS) networks have been enhanced with the new features of the Automatically Switched Optical Network (ASON, [1]) architecture, able to dynamically set up and release circuits across the optical network in a few hundreds of milliseconds. The enabling entity to these functionalities is a common control plane, typically implemented by means of the Generalized Multi-protocol Label Switching (GMPLS, [2]) protocol set. On the one hand, OCS networks allow efficient and QoS compliant data transmission for long-lived flows. On the other hand, they offer poor bandwidth usage and reduced adaptation to bursty data traffic patterns. Note that OCS networks provide a very coarse bandwidth granularity, in the order of a full wavelength. Besides, connection set up and release may perform rather slow when connection holding times are very short, resulting in a high signaling overhead.

Newly proposed switching paradigms like Optical Packet Switching (OPS, [3]) and Optical Burst Switching (OBS, [4]) exploit the statistical multiplexing directly in the optical domain, allowing fine sub-wavelength granularity. Nonetheless, realization complexities arise in both technologies compared to OCS. This is especially noticeable in OPS, which makes it difficult to deploy in the foreseeable future. To lessen OPS technology requirements, OBS networks lie between OCS and OPS, trying to combine benefits of both paradigms while minimizing their disadvantages.

However, absolute QoS guarantees is still an important yet challenging issue in OBS networks. Moreover, from an economic viewpoint, a pure OBS network needs a high number of expensive burst switch ports, which may even reach technological limits [5]. Supported by these arguments, hybrid OBS/OCS networks have appeared as an efficient and cost-effective solution for future optical transport network infrastructures. These networks employ OBS and OCS switching technologies simultaneously. For instance, in the hybrid OBS/OCS scenarios proposed in [6], [7], [8], OBS plays an important role in efficiently carrying short-lived flows and best-effort bursty traffic, whereas long-lived data transmission with QoS guarantees is supported on OCS. It would be the task of the ingress router to choose the most appropriate transport service for the incoming data flow, based on either QoS requirements or flow duration (e.g., see [7], [9]). Other degrees of integration and interaction between OBS and OCS technologies in hybrid OBS/OCS networks are described in [10].

This paper focuses on optimizing OBS layer performance in hybrid OBS/OCS network scenarios. With such purposes, we address the feasibility of using resource state information for routing decisions at the OBS layer. As will be shown, the state information dissemination mechanism becomes crucial to overall OBS layer performance. We avoid the use of flooding-based link state protocols such as OSPF-TE [11], due to their high complexity and slow convergence, which does not match OBS operation time-scales. In fact, to mitigate inaccuracies due to propagation delays, we only contemplate neighboring nodes state information rather than trying to get a whole network view. This information will be included in either the

Burst Control Packet (BCP) or in the Link Management Protocol (LMP, [12]) Hello messages exchanged over the GMPLS-enabled OCS control layer. Knowing the state of surrounding neighbors, we propose a Preventive Deflection Routing (PDR) protocol, which reacts to highly loaded downstream node situations by preventively deflecting bursts towards a less loaded neighbor. Looking at the results, we conclude that quite updated neighboring nodes state information is enough to provide significantly better results than classic Deflection Routing (DR, [13]).

The rest of this article continues as follows. Section 2 discusses state information dissemination mechanisms in hybrid OBS/OCS networks. Section 3 introduces the PDR concept as well as the applied routing heuristics. Section 4 illustrates the scenario under study. Section 5 presents the obtained results. Finally, section 6 concludes the paper.

II. State Information Dissemination in Hybrid OBS/OCS Networks

In dynamic OCS networks, link-state information is typically disseminated by means of OSPF-TE, which enables Constrained Shortest Path First (CSPF) route calculations with a whole network view, that is, considering only those currently available resources. These CSPF calculations become accurate whether connection Inter-Arrival Times (IATs) and Holding Times (HTs) are significantly higher than the time it takes OSPF-TE to flood resource state changes. While this is fulfilled in OCS, it does not happen in OBS. In OBS networks, IATs stay in the order of μs, whereas HTs range from tens of μs to several ms. This would lead to totally outdated link-state information, useless for path computation. Indeed, experimental measurements in the ASON/GMPLS CARISMA Test-bed showed that OSPF-TE takes up to few seconds to disseminate resource state changes [14]. This paper proposes the use of partial network state information at the OBS layer to minimize the impact of propagation delays in state information accuracy. Herein, we do not try to get a whole network view as in OCS networks, but we only contemplate neighboring nodes state information.

It is our objective to gather general but relevant state information, rather than dealing with a description of each wavelength occupancy all through the time. In this way, we avoid to excessively overload the control network and we simplify deflection routes computation, which should be done in a very restrictive time-budget at intermediate OBS nodes [15]. Specifically, we consider the ratio of currently allocated output ports in the node (i.e., those currently transmitting a burst) as a metric. Note, that such information does not imply a large overhead, as a simple Byte is enough to represent an occupancy percentage from 0 to 100. In this section, two alternative state information dissemination approaches are introduced to keep surrounding neighbors informed of the gathered node state information.

The considered control plane scenario for hybrid OBS/OCS networks is similar to the one presented in [16]. We assume a hybrid control plane, composed of a specific OBS control layer and an OCS control layer on top implemented by means of GMPLS. The OBS control layer supports the BCP transmission regarding the OBS layer signaling. Recall, that this one must share the same resources and topology as the OBS layer data plane, as bursts and BCPs must keep a strict time relationship in OBS [4]. In turn, the OCS control layer is responsible for the set up, maintenance and release of circuits over the OCS data plane. In contrast to the OBS control layer, this one could be implemented following a different topology than the OCS data plane, even supported over a separated network.

A. Node state information in the BCPs

The first approach targets at updating neighbor state information in the OBS time-scales by inserting node state information in the BCP. Once a burst has to be sent to a given downstream node, current node state information is included in the BCP. We introduce a new field in the BCP called *Occupancy_Rate_Byte*, which contains the ratio of allocated output ports in the node expressed as a percentage from 0 to 100. In this way, upon BCP arrival, the downstream node can retrieve the information of the upstream node state from the BCP and store it. Next, if it is still a transit node, it can include information of its own current state in the BCP, so that it can be forwarded to the the next node.

As the result, provided that traffic is sufficiently distributed in the network (one of the OBS motivations), OBS nodes maintain quite accurate neighbor state information, keeping pace with OBS dynamics (updates from neighboring nodes come in the order of burst IAT intervals). Note, that this solution is not only applicable to hybrid OBS/OCS networks but also to pure OBS ones, as no cross-layer interaction is required.

B. Node state information in the LMP Hello messages

The second approach works in the OCS time-scales, reducing the introduced control overhead in the first mechanism. The idea is based on transmitting the nodes state information over the OCS control layer in the Hello messages of the LMP protocol, used to maintain the connectivity of the control channels established between neighboring nodes.

As studied in [14], control channel connectivity maintenance becomes critical in GMPLS-controlled networks, given the flexibility of the GMPLS control plane to be physically decoupled from the data plane [2]. With such purposes, LMP standardization proposes Hello messages to be exchanged between neighbors every 150 ms (i.e., *HelloInterval*), so that if no Hello is received from the neighbor along a 450 ms period (i.e., *HelloDeadInterval*), the control channel is declared down. This would assure, under normal control channel operation, that nodes receive state information updates every 150 ms.

Note, however, that in this case operating time-scales significantly differ from those of OBS. Hence, we propose to exchange averaged node state estimations rather than current

information as in the previous solution. Specifically, the included information is an averaged node output port occupancy percentage along the last *HelloInterval* period. To this end, current node occupancy percentage samples are polled every 10 ms. Thus, if a node has to send a new Hello mesage to a neighbor, those samples gathered in the last *HelloInterval* are averaged and included in a newly defined *Occupancy_Rate* LMP object in the Hello message.

III. Preventive Deflection Routing

In a classic DR scenario [13], contention situations are solved by reactively deflecting contending bursts to an alternative output fiber of the node, following the shortest path to the destination. However, if a burst reaches a node and all output ports (those which would lead to the destination) are already reserved, the burst is inevitably lost. In this context, the knowledge of neighboring nodes state permits deflection decisions to be not only reactive due to contention but also preventive. In fact, knowing in advance that the downstream node is highly loaded, burst losses could be avoided by preventively deflecting bursts towards a less loaded neighbor, even though no contention occurs on the direct downstream link.

In the here presented PDR, preventive deflections are triggered whether the occupancy rate L_v of the downstream node v (i.e., the ratio of currently allocated output ports, $0 \leq L_v \leq 1$) exceeds a certain threshold L_{th}. Notice, however, that excessive deflections impose additional traffic in the network, as deflection routes are usually longer than the primary ones. In addition, albeit downstream node load exceeds L_{th}, the burst might not be dropped there. Imagine that L_v equals to 0.75, whereas L_{th} is set to 0.7. This would trigger a preventive deflection. Nonetheless, there is still a 25% of idle ports that may be allocated for the burst. Further attention requires the inaccuracy of the stored neighbor state information, which may yield unnecessary preventive deflections and vice versa.

To control the increased traffic load due to excessive deflections, we introduce in PDR a parameter called p_{pdr}, so that $p_{pdr} = P(A|L_v > L_{th})$, where A identifies the event of a burst to be preventively deflected. Observe that $p_{pdr} = 1.0$ fosters preventive deflections to be carried out provided that $L_v > L_{th}$. Contrariwise, $p_{pdr} = 0.0$ allows deflections to be only reactive, as in classic DR. Mention, that preventive deflections take no sense if the downstream node is the destination one, as the burst is delivered in the next node and no further contention can be experienced. Hence, only reactive deflections are permitted in such a situation.

The remainder of this section concentrates on the presentation of the applied heuristics to reroute bursts either upon preventive or reactive deflection actions.

A. Proposed routing heuristics

We model the network as a graph $G = (V, E)$, where V represents the set of nodes and E the set of physical WDM links. For this network, a pool of wavelengths per link $\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_W\}$ is defined. Moreover, for each node v, $v \in V$, we denote its neighbor set as $\Gamma(v)$, so that $\Gamma(v) = \{v'|(v, v') \in E\}$. From now on, we name available neighbors to such neighbors towards which available resources would be found when trying to send a given burst.

Algorithm 1: PDR Routing Algorithm

```
Input: G = (V,E), s, d
Output: P = Deflection Route between s and d
G' = G;
S = Γ(s);
W_SP = ∞;
for v = next (S) do
    if HasResources(v) then
        W_v = L_v · ConstrainedDist(s,d,v);
        if W_v < W_SP then
            W_SP = W_v;
        else
            G' = G' - (s,v);
        end
    else
        G' = G' - (s,v);
    end
end
if W_SP < ∞ then
    P = Dijkstra (G',s,d);
else
    P = φ;
end
```

The idea of the proposed heuristics is to exploit the stored neighboring nodes state information to provide the most appropriate deflection route to the destination. Two extreme situations come up to this goal. On the one hand, we could force bursts going through the least congested available neighbor, thus ensuring in most situations that the burst would not be dropped on the following hop. This strategy, however, would dramatically increase the offered traffic in the network, since deflection routes would not be selected following a shortest path strategy, undoubtedly leading to an undesired overall network performance. On the other hand, we could select deflection routes only taking into account the distance to the destination. This would result in a similar behavior as classic DR, making the stored state information useless. Hence, the selected deflection route should lie between these two extreme situations.

In Algorithm 1, s and d denote the source and destination nodes respectively and W_{SP} stands for the cost associated to the most appropriate route found so far. Note that W_{SP} is initialized to ∞, as no deflection route is yet selected when the algorithm starts. The function $HasResources(v)$ checks whether available resources exist in the output port connected to the adjacent neighbor v. Besides, $ConstrainedDist(s, d, v)$ returns the distance (e.g., in number of hops) from node s to to node d when the route is constrained to go through the downstream node v.

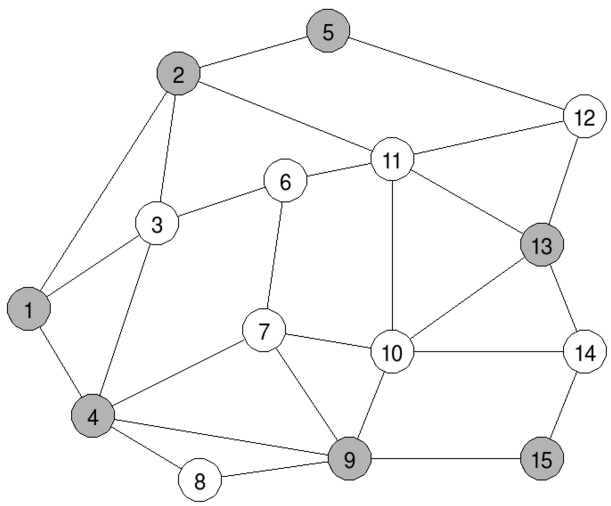

Fig. 1. Scenario under study; traffic demands are uniformly distributed among source/destination nodes 1-2-4-5-9-13-15.

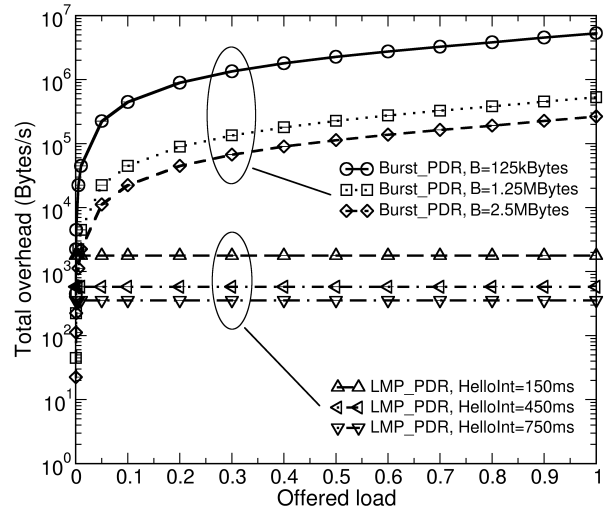

Fig. 2. Overhead comparison between Burst_PDR and LMP_PDR as a function of the offered load per node.

The rationale behind the routing heuristics applied in PDR is to find a deflection route to the destination as short as possible, avoiding those highly loaded downstream neighbors as well. With such purposes in mind, for each neighbor v, resource availability to send the burst is checked. If no resources are found, the link (s, v) is directly excluded from a simplified network graph $G' = (V, E)$, which will be finally used to obtain the deflection route by means of a Dijkstra shortest path algorithm. Not being the case, the algorithm checks if the cost W_v of the deflection route from s to d going through v is lower than W_{SP}. Specifically, we set the route cost as $L_v \cdot ConstrainedDist(s, d, v)$, so that we penalize both longer routes and highly loaded neighbors. If $W_v \geq W_{SP}$, (s, v) is also excluded from $G' = (V, E)$, as a shorter route has been previously found. Otherwise, the algorithm sets $W_{SP} = W_v$ and it proceeds with the following neighbor, if any. Finally, the Dijkstra shortest path algorithm is applied to $G' = (V, E)$ to find the deflection path P. Nevertheless, if $W_{SP} = \infty$, $P = \phi$ is directly returned, as it means that no neighbor with available resources was found.

IV. Scenario under Study

With evaluation purposes, we simulated the KL-network [17] depicted in Fig. 1, assuming that links carry 16 bidirectional wavelengths at 10 Gbps. Besides, link lengths were chosen to be 200 km.

For the traffic characteristics, we consider that bursts depart from each node following a Poisson process with a mean burst IAT equals to $1/\lambda_b$. Particularly, bursts are uniformly distributed to all the remaining nodes of the network, so that the probability of a departing burst to be sent to any remainder node i is $P_i = 1/(N-1)$, where N equals to the number of nodes in the network. Burst size B follows an exponential distribution with mean 1.25 MBytes in our scenario. This leads to an exponentially distributed burst length τ with mean $1/\mu_b$ = 1 ms, that is, the time needed to transfer a 1.25 MBytes burst onto a 10 Gbps link.

As mentioned in Section 2, a hybrid control plane is implemented. A GMPLS protocol stack is running over the OCS control layer, whereas the OBS control layer provides the transmission medium for the BCPs between neighboring nodes. Particularly, control channels at the OCS control layer are maintained by means of LMP, whose default *HelloInterval* value has been set to 150 ms.

Finally, regarding hardware devices, we assumed that OBS nodes are equipped with full wavelength conversion, a non-blocking switching matrix and an enough number of add/drop ports. The BCP processing time and the matrix switching time were set to 10 μs and 2.5 μs respectively. Moreover, as in [18], a Fiber Delay Line (FDL) was placed at each input port of the node, which compensates the processing delay incurred by the BCP at the control unit. Note, that such architecture enhances fairness in resource allocation for the bursts, as offset time is not decreased along the path [19]. In addition, due to the constant offset times no void-filling is needed, as no overtaking situation occurs.

V. Simulation Results

This section focuses on the validation of the proposed PDR protocol, depending on whether resource state dissemination is achieved by using either the BCPs or the LMP Hello messages. Both solutions are hereafter referred as Burst_PDR and LMP_PDR respectively. To this objective, the reduction of control overhead in LMP_PDR in front of Burst_PDR at expenses of sacrificing state information accuracy is firstly quantified. As a further step, burst loss probability figures of Burst_PDR and LMP_PDR are compared to the one obtained with classic DR. It is worth mentioning that in all results presented in this section, $L_{th} = 0.7$ has been set to trigger a preventive deflection.

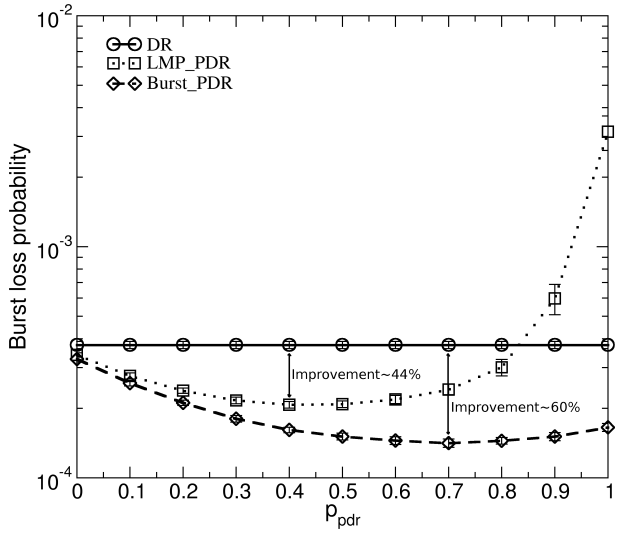

Fig. 3. Burst loss probability as a function of p_{pdr} for Burst_PDR, LMP_PDR and DR.

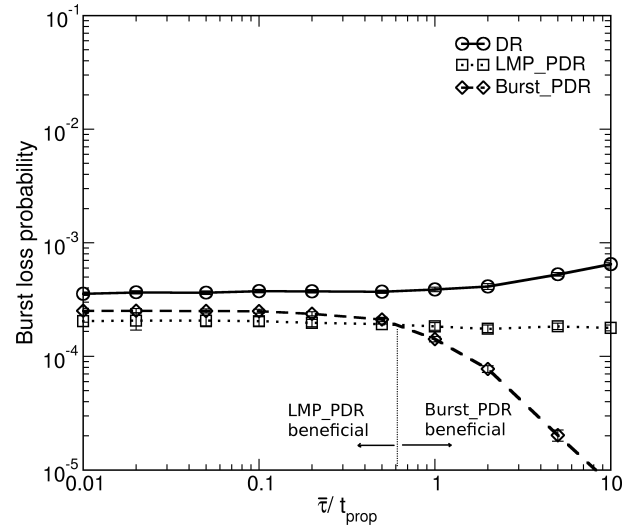

Fig. 4. Burst loss probability as a function of $\overline{\tau}/t_{prop}$ for Burst_PDR, LMP_PDR and DR.

A. Overhead analysis of Burst_PDR and LMP_PDR

As mentioned before, resource state dissemination in the LMP Hello messages was an interesting solution from the overhead point of view. Note that this solution works in the OCS time-scales, thus much less state information is expected to be exchanged between neighboring nodes. Further benefits from this solution in front of using the BCPs lies in the fact that the introduced overhead becomes independent of the burst IAT values in the network. Interesting enough, the total amount of transmitted state information only depends on the chosen LMP *HelloInterval*, typically much higher than burst IATs.

For overhead quantification purposes, Fig. 2 depicts the total introduced overhead in the network in Bytes/s as a function of the offered load per node, depending on whether Burst_PDR or LMP_PDR is applied. Different mean burst size values are contemplated in Burst_PDR. In fact, under Poisson burst departures, the total offered load per node equals to λ_b/μ_b (i.e., HT/IAT). Hence, for a given load, different mean burst size values lead to different burst IATs, thus appreciating IAT impact on total introduced overhead. In LMP_PDR, different *HelloInterval* values have been also evaluated. Specifically, 1 overhead Byte per BCP has been considered in Burst_PDR, enough to represent a percentage value ranging from 0 to 100. Conversely, 5 overhead Bytes are counted per Hello message in LMP_PDR, that is, 4 Bytes of the LMP object header [12] plus 1 Byte field to carry the state information. Initially, $p_{pdr} = 1.0$ was set in both schemes.

Looking at the results, an overhead reduction between two and three orders of magnitude for LMP_PDR against Burst_PDR can be appreciated. While LMP_PDR introduces around 1 kByte/s overhead in the network under study, Burst_PDR approximately requires 1 MByte/s. As a matter of fact, the introduced control overhead in Burst_PDR is highly dependant on both the offered load and mean burst size, so that the larger the bursts, the lower the control overhead, since burst IAT values are also increased. Contrarily, LMP_PDR provides the expected constant behavior as the offered load increases. Moreover, the overhead differences for the alternative *HelloInterval* values remain quite low.

B. Performance comparison amongst Burst_PDR, LMP_PDR and classic DR

The study in Fig. 3 describes the performance of Burst_PDR and LMP_PDR as a function of p_{pdr}. For the results, an offered load per node equal to 0.7 has been assumed, which leads to burst loss probabilities in the typical OBS operating range (burst losses ranging from 10^{-3} to 10^{-6}).

As can be seen, the results are in line with the arguments presented in section 3. First of all, whether $p_{pdr} = 0.0$, the improvements of both schemes in comparison to classic DR are marginal. In fact, in such a case, only reactive defections are allowed. Though burst losses at the first hop of the defection route are minimized, we usually introduce quite longer routes. More important, we do not avoid such situations where bursts are sent to highly loaded neighbors as long as no contention exists on the downstream link. This demonstrates the necessity of preventive defection decisions. Nonetheless, it is worth to highlight that $p_{pdr} = 1.0$ also worsens network performance. Since the stored neighbor information is sometimes inaccurate, triggered preventive deflections are in some occasions unnecessary. Hence, the amount of traffic in the network is increased, as deflection routes are usually longer than the primary ones. This is especially evident in LMP_PDR, which works in the OCS time-scales and the preventive deflection decisions are based on averaged estimations.

In particular, LMP_PDR best results are achieved for $p_{pdr} = 0.4$, leading to a 40% improvement to classic DR when only 40% of the total possible preventive deflections are allowed. In contrast, noticeably better results are achieved by Burst_PDR, due to the more accurate maintained state information. In this case, around 60% improvement to classic DR is obtained

when $p_{pdr} = 0.7$. These obtained improvements have been calculated as $100 \cdot (PLoss_{DR} - PLoss_{PDR})/PLoss_{DR}$.

Until now, we have considered an exponentially distributed burst length with mean $\overline{\tau} = 1$ ms. Fig. 4 gives insight into how Burst_PDR and LMP_PDR behave as $\overline{\tau}$ varies, which is also compared to the performance of classic DR. To this end, we fix an offered load per node equal to 0.7 and we plot burst loss probability as a function of $\overline{\tau}/t_{prop}$, being t_{prop} the link propagation time in the network (i.e., 1 ms in our scenario). According to the previous results, p_{pdr} has been set to 0.7 and 0.4 in Burst_PDR and LMP_PDR respectively.

As shown, both Burst_PDR and LMP_PDR outperform classic DR in all the evaluated range. Therein, we observe an almost constant behavior of LMP_PDR and classic DR with the burst length. This is not the case, however, of Burst_PDR. Note that Burst_PDR uses current node state information for routing purposes and preventive deflection decision. In this context, $\overline{\tau}$ plays an important role in the accuracy of the stored resource state information. If bursts are short, nodes state varies quickly thus being state information rapidly outdated. Contrariwise, if bursts are long-lived, stored state information remains updated for a longer time. As can be observed, for very short bursts, almost the same performance is achieved by both Burst_PDR and LMP_PDR, behaving the latter one slightly better than the former. In fact, in this range, it is even better to use averaged estimations rather than trying to use highly variable current state information. However, towards higher $\overline{\tau}$ values, a cross-over point exists at $\overline{\tau}/t_{prop} = 0.6$. From then on, Burst_PDR starts to behave efficiently. Note that Burst_PDR obtains more than one order of magnitude improvements for $\overline{\tau}/t_{prop}$ values greater than 2 (i.e., $\overline{\tau} > 2$ ms). It is noteworthy, that such values are totally suitable for OBS. For example, the Open Grid Forum (OGF) considers milliseconds' burst lengths for evolving Grid-OBS network architectures [20].

Therefore, Burst_PDR applicability becomes appropriate when bursts are large mainly for two reasons. First, burst loss probability is drastically reduced in comparison with classic DR and LMP_PDR. Second, as illustrated in previous subsection, burst IAT values are proportional to burst length. Therefore, the longer the bursts, the larger the IATs are, which decreases the introduced control overhead. Contrariwise, short burst length fosters the applicability of LMP_PDR, not only due to the lower burst loss probability, but also due to the significantly reduced control overhead, independent of the incoming traffic dynamics.

VI. Concluding Remarks

In this paper, we introduced a load-based preventive deflection routing protocol for hybrid OBS/OCS networks, as well as two different state information dissemination schemes. As a first approach, we proposed state information to be carried in the BCPs, which keeps pace with OBS traffic dynamics. Alternatively, to reduce the introduced control overhead, we proposed state information to be included in the Hello messages of the GMPLS LMP protocol, running on the OCS control layer. The obtained results assess the applicability of the proposed PDR protocol. Particularly, we identified that LMP based resource state dissemination becomes favorable for short and medium sized bursts, providing better burst loss probability figures while introducing a low control overhead. In contrast, for large burst sizes, Burst_PDR starts to behave efficiently, showing more than one order of magnitude burst loss reduction against classic DR.

Acknowledgment

The work described in this paper was carried out with the support of the BONE Project ("Building the Future Optical Network in Europe"), a Network of Excellence funded by the European Commission through the 7th ICT-Framework Programme. Moreover, the authors thank the support from the Spanish ministry through the project ENGINE (TEC2008-02634).

References

[1] A. Jajszczyk, "Automatically Switched Optical Networks: Benefits And Requirements", IEEE Commun. Magazine, Feb. 2005.
[2] E. Mannie, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", IETF RFC 3945, Oct. 2004.
[3] S. Yao, B. Mukherjee, S. Dixit, "Advances in photonic packet switching: an overview", IEEE Commun. Magazine, Feb. 2000.
[4] Y. Chen, C. Qiao, X. Yu, "Optical Burst Switching: A new area in Optical Networking Research", IEEE Network Magazine, Jun. 2004.
[5] H. Buchta et. al., "Limits of effective throughput of optical burst switches based on semiconductor optical amplifiers", in Proc. of OFC, Mar. 2003.
[6] H.L. Vu et. al., "Scalable performance evaluation of a hybrid optical switch", IEEE/OSA Journal of Lightwave Technology, Oct. 2005.
[7] G. M. Lee et. al., "Performance evaluation of an optical hybrid switching system", in Proc. of IEEE GLOBECOM, Dec. 2003.
[8] C. Xin, C. Qiao, Y. Ye, S. Dixit, "A hybrid optical switching approach", in Proc. of IEEE GLOBECOM, Dec. 2003.
[9] G. Zervas et. al., "A Hybrid Optical Burst/Circuit Switched Ingress Edge Router for Grid-enabled Optical Networks", in Proc. of BROADNETS 2006, Oct. 2006.
[10] C.M. Gauger et. al., "Hybrid optical network architectures: bringing packets and circuits together", IEEE Commun. Magazine, Aug. 2006.
[11] D. Katz et. al., "Traffic Engineering (TE) Extensions to OSPF Version 2", IETF RFC 3630, Sept. 2003.
[12] J. Lang, "Link Management Protocol", IETF RFC 4204, Oct. 2005.
[13] S. Yao et al., "A Unified Study of Contention-Resolution Schemes in Optical Packet-Switched Networks", IEEE/OSA Journal of Lightwave Technology, Mar. 2003.
[14] J. Perelló et. al., "An Analytical Study of Control Plane Failures Impact on GMPLS Ring Optical Networks", IEEE Commun. Letters, Aug. 2007.
[15] N. Barakat, T.E. Darcie, "Control-Plane Congestion in OBS Networks", in Proc. of BROADNETS 2006, Oct. 2006.
[16] P. Pedroso et. al., "An interoperable GMPLS/OBS Control Plane: RSVP and OSPF extensions proposal", in Proc. of CSNDSP, Jul. 2008.
[17] M. Kodialam, T. Lakshman, "Integrated dynamic IP and wavelength routing in IP over WDM networks", in Proc. of IEEE INFOCOM, Apr. 2001.
[18] Y. Xiong, M. Vanderhoute, H.C. Cankaya, "Control architecture in optical burst-switched WDM networks", IEEE Journal on Selected Areas in Communications, Oct. 2000.
[19] M. Klinkowski, D. Careglio, J. Solé-Pareta, "Offset-Time-emulated OBS control architecture", in Proc. of ECOC 2006, September 2006.
[20] "Grid Optical Burst Switched Networks (GOBS)", Open Grid Forum (OGF), GHPN Group, Informational track draft, Oct 2007.