Adaptive Routing and Contention Resolution Approaches for OBS Networks with QoS differentiation

Abdeltouab Belbekkouche[1], Abdelhakim Hafid[1], Michel Gendreau[2]

[1]Network Research Laboratory
University of Montreal, Montreal, Canada
{belbekka, ahafid}@iro.umontreal.ca

[2]CIRRELT
University of Montreal, Montreal, Canada
michel.gendreau@cirrelt.ca

***Abstract*—Optical Burst Switching (OBS) is a promising switching paradigm for the next generation Internet. A buffer-less OBS network can be implemented simply and cost-effectively without the need for either wavelength converters or optical buffers which are, currently, neither cost-effective nor technologically mature. However, this type of OBS networks suffers from relatively high loss probability caused by wavelength contentions at core nodes. This issue could prevent or, at least, delay the adoption of OBS networks as a solution for the next generation optical Internet. To enhance the performance of buffer-less OBS networks, we propose two approaches: a) a proactive multi-path approach, called Reinforcement Learning-based Alternative Routing (RLAR), that tries to prevent wavelength contentions; and b) an approach, called Integrated Reinforcement Learning-based Routing and Contention Resolution (IRLRCR) that deals with wavelength contentions proactively and reactively conjointly. We also show relative QoS differentiation capability of the proposed schemes. Simulation results show the ability of the proposed approaches to reduce loss probability and to provide Quality of Service (QoS) differentiation.**

Keywords: Multi-path routing; Deflection Routing; Optical Burst Switching; Unsupervised learning; Wavelength Division Multiplexing; QoS differentiation

I. INTRODUCTION

WDM (Wavelength Division Multiplexing) is an attractive technology to support the huge amount of data required by the future optical Internet. It uses the potential capacity in optical fibers that contains many wavelengths able to carry many Gbps by using statistical multiplexing. This potential requires good switching technology to efficiently exploit it. OBS (Optical Burst Switching) [1, 2] is a good candidate switching paradigm to fill this need. It has received an increasing interest from researchers over the last years.

In OBS networks, data packets with the same destination are aggregated in bursts of variable lengths at the ingress node, this is called *Burst Assembly*. After burst assembly, a *Control Packet* (also called *Burst Header Packet*) is sent, using a dedicated control wavelength, from source to destination in order to reserve the required resources along a light path. This control packet is subject to *Optical-Electronic-Optical* (OEO) conversions at each core node (OBS switch) where it receives an appropriate processing. After a delay called *Offset Time* (OT), the corresponding data burst is sent, on one of the data wavelengths, through the same lightpath without any buffering requirement inside the OBS network.

A major issue in OBS networks is wavelength contention which is the main cause of burst losses; this may result in a high burst loss probability (defined as the rate of bursts lost in the OBS network core nodes) causing a considerable performance degradation of OBS networks. A contention arises when two or more bursts intend to take the same output fiber, on the same wavelength, at the same time. There exists mainly two kinds of approaches to deal with wavelength contention: *reactive approaches* (e.g. deflection routing where only one of the bursts involved in a contention is routed to its primary output fiber, whereas each of the other bursts is switched to an alternative outgoing fiber) and *proactive approaches* (e.g. multi-path routing). Whereas reactive approaches try to resolve contentions after they occur in a network core node (generally, based on local information of this node), proactive approaches attempt to prevent contentions, and consequently burst losses, from occurring.

In this paper, we consider an OBS network without wavelength converters (which are complex, expensive and not yet technologically mature) and without Fiber Delay Lines (FDLs) (which are inflexible). This makes our network under study cost-effective and simple to implement with the existing technology from techno-economic point of view. We propose two approaches to reduce the loss probability of the buffer-less OBS network. The first approach, called *Reinforcement Learning-based Alternative Routing* (RLAR), is a proactive approach that aims to establish a suitable load balancing in OBS networks to reduce burst losses. In RLAR, control packets try to find an optimal path, in terms of both burst loss probability and end-to-end delay. RLAR is scalable and well adapted to dynamic changes in the network state and topology. The second approach, called *Integrated Reinforcement Learning-based Routing and Contention Resolution* (IRLRCR), is the combination of RLAR with a reactive approach called *Reinforcement Learning-based Deflection Routing Scheme* (RLDRS) [3] to deal with wavelength contentions reactively (using RLDRS) and proactively (using RLAR) conjointly. Also, we show the ability of RLAR and IRLRCR to provide Quality of Service (QoS) differentiation for both loss-sensitive applications and delay-sensitive

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

applications.

The remainder of this paper is organized as follows. Section II presents related work on multi-path routing. Section III describes the proposed multi-path routing approach (RLAR). In Section IV, we describe the integrated approach (IRLRCR). Section V discusses the ability of RLAR and IRLRCR to provide relative QoS differentiation. In Section VI, we present simulation results. Finally, Section VII concludes the paper.

II. RELATED WORK

In the literature of OBS networks, we find a number of contributions which have studied alternative routing [4-8]. These contributions use multi-path routing where a set of paths is a priori known between each source and destination pair of nodes. In this case, the choice of the routing path to a given destination is performed at the source node. Hence, the problem is what information to use, how to update it and how to determine the optimal path given this information. The authors in [5] consider several metrics (e.g. link utilization, path end-to-end delay or burst loss probability, etc.) to evaluate the level of congestion in a given path. The choice of the optimal path could be based on pure strategy (if a single metric is considered) or hybrid (if two or more metrics are combined). The authors conclude that, in general, path switching outperforms shortest path routing that is usually used in OBS networks. In [7], paths from the same source to the same destination are assigned priorities; bursts are sent using a path that has the highest priority. Each time a data burst is sent on a given path, the source node receives a feedback which indicates either the success or the failure of the burst transmission so that it can update path priorities. To accelerate the rate of priorities updating process, each time a data burst is sent, search packets are sent on the other paths leading to the same destination (using control wavelengths) in order to receive feedbacks indicating whether transmitting the data burst on the other paths would be successful or not. We note that search packets may cause a considerable control overhead. The authors in [6] propose a congestion avoidance-based scheme in which each network core node measures the load on each one of its output links and sends, periodically, this information to all of the network edge nodes; these nodes use the load information to avoid congested paths. The authors propose another scheme that considers a weighted function based on congestion and hop count (between source and destination) to select paths that improve performance in terms of burst losses and delay. In [8], a passive probing on sub-optimal paths is used; the source node sends a small fraction of its traffic on these paths while keeping the probabilities of choosing them very low and analyzes feedback ACK and NACK packets during a time sliding window to determine an optimal path. A burst pipelining scheme is also proposed to guarantee in-order burst delivery. More recently, the authors in [4] proposed to put reinforcement learning agents using Q-learning at each ingress edge node. An agent chooses for every burst, to be transmitted, the optimal path in terms of burst loss probability and use ACK and NACK feedback packets to update its appreciation of paths. In this paper, RLAR uses learning agents at both edge and core nodes to choose optimal links rather than an optimal path; this results in a finer granularity path selection. In addition, RLAR considers both burst loss probability and end-to-end delay as metrics for the path selection process. Furthermore, control overhead is considerably reduced since RLAR uses only one-hop feedback packets of very small size.

III. THE PROPOSED MULTI-PATH ROUTING APPROACH

In this section, we present our distributed Q-learning-based routing approach. First, we present a Q-learning based scheme that will be used at each OBS network node to determine the optimal output link to forward an incoming burst towards its destination. Then, we discuss issues related to reinforcement learning, namely, exploration versus exploitation and convergence; and we present overhead analysis of the proposed scheme. Finally, we present an algorithm that guarantees the computation of loopless paths.

The objective of our proposal is to find optimal paths to route bursts in the context of multi-path routing. The key difference between our approach and existing multi-path routing solutions, is that our approach distributes the decision process of selecting optimal paths on all the nodes in the OBS network; this results in (a) a better selection of paths thanks to more accurate routing information. Indeed, each node can measure its local congestion with more accuracy (compared with end-to-end measurement); and (b) better robustness: link failure will be detected more rapidly (compared to centralized/edge based detection). The selection process does not only take into account burst losses (like existing schemes) but also end-to-end delay. The integration of learning techniques in the proposed approach makes it adapt better to changes in the state of the network.

A. *The routing scheme*

We propose to make the path selection process distributed on all the nodes in the network, i.e., any node in the network has to decide which link a burst should take to reach its destination. Each node in the network has a learning agent that learns the optimal output link to forward a burst to a given destination at a given time. A learning agent uses a lookup table, called *Q-table*, to store values (called Q-values) representing its appreciation of an output link with respect to a given destination. This appreciation takes into consideration both burst loss probability and delay (in terms of hops) experienced by bursts from the current node towards the destination through a chosen output link. Hence, each entry in the node's Q-table is indexed by the pair *(destination, neighbor)*; the computation of Q-value will be described later in the section. When a learning agent receives an incoming burst, it decides to forward this burst to a neighbor with the highest Q-value. For example, if a node x receives a data burst with destination d, it forwards that burst to its neighbor y determined in (2):

$$y = \arg\max_{z \in N(x)} Q_x(d, z) \tag{2}$$

where *arg max* stands for argument of the maximum (namely, the neighbor with the maximum Q-value with respect to a given destination), $Q_x(d,z)$ is the Q-value that corresponds to neighbor z and destination d, and $N(x)$ is the set of neighbors of node x.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

Initially, we assume that the loss probability of each output link in the OBS network is null. To initialize the Q-table of a node, say x, each Q-value corresponding to a given destination, say d, and a given neighbor, say y, is computed based on the shortest path delay (the number of hops) between x and d where the first hop neighbor is y. Thus, we ensure that if loss probability is very low or negligible, RLAR converges to *Shortest Path Routing*.

Whenever a node sends/forwards a control packet to a neighbor, it receives a feedback packet, from that neighbor, that it uses to update that neighbor's (and corresponding destination) entry in the Q-table. For example, in Fig. 1, when node x sends a control packet with destination d to its neighbor y, it receives a feedback packet from y which contains a numerical value f_{yx} defined in (3):

$$f_{yx} = Q_y(d,z) \cdot D_y(d,z) \qquad (3)$$

where node z and $Q_y(d,z)$ are determined by node y using (2), and $D_y(d,z)$ is the delay between node y and node d through neighbor z (the delay is the number of hops of the shortest path between node z and node d). We assume that each node in the OBS network, knows the number of hops of the shortest path between it and each destination through each of its neighbors. The multiplication of $Q_y(d,z)$ by $D_y(d,z)$ in the feedback value f_{yx} is necessary to eliminate the delay factor already considered in $Q_y(d,z)$; this delay is not useful for the destination node of the feedback packet (node x in this case), that will use its own delay to the destination node d (see (4)).

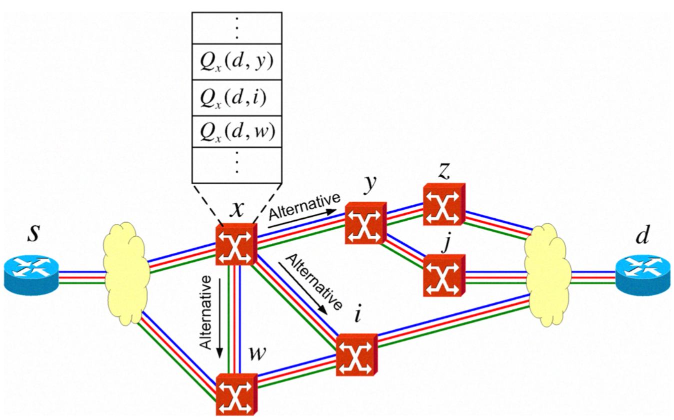


Figure 1. An example that shows a part of node x's Q-table.

Upon receipt of the feedback packet, node x updates its Q-table (Q-value) as shown in (4):

$$Q_x(d,y) \leftarrow Q_x(d,y) + \alpha \cdot ((f_{yx} \cdot (1 - B_{xy}) / D_x(d,y)) - Q_x(d,y)) \qquad (4)$$

where $0 < \alpha \leq 1$ is the learning rate, B_{xy} is the burst loss probability on the output link from node x to node y. It is measured using a time sliding window; at the end of each time window of duration τ, B_{xy} is calculated as shown in (5):

$$B_{xy} = \begin{cases} \dfrac{Drop_{xy}}{Drop_{xy} + Sent_{xy}}; & \text{If } Sent_{xy} + Drop_{xy} > 0 \\ 0; & \text{If } Sent_{xy} + Drop_{xy} = 0 \end{cases} \qquad (5)$$

where $Drop_{xy}$ and $Sent_{xy}$ are the number of dropped bursts and successfully transmitted bursts through the output link from node x to node y during the last time window, respectively.

The idea behind (4) is to estimate the probability that a burst will be dropped along the path from x to d through y. Indeed, assuming that link drop probabilities are independent, the probability that a burst will be dropped through a path P consisting of nodes $n_1, \ldots, n_{|P|}$ (noted b_p), is calculated using (6):

$$b_p = 1 - \prod_{1 \leq i \leq |P|-1} (1 - B_{n_i n_{i+1}}) \qquad (6)$$

In (4), we introduce the delay (number of hops) of the considered path to make a tradeoff between loss probability and delay. Thus, shorter paths are preferred over longer paths when the longer path's improvement in terms of loss probability is not substantial. Fig. 2 shows the operation of RLAR at an OBS core node.

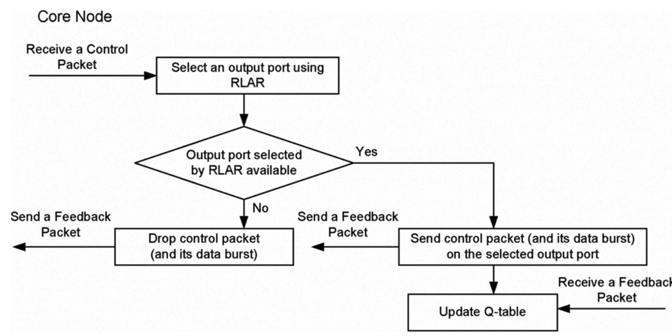


Figure 2. The operation of RLAR at an OBS core node.

B. *Exploration, convergence and overhead analysis*

Exploration is an important concept in reinforcement learning. Indeed, the proposed routing scheme aims to exploit what a learning agent has learnt before (from its interaction with the environment) by selecting the neighbor with the highest Q-value in the Q-table, to forward an incoming burst; this is called *exploitation*. Nevertheless, an exploration policy is required to check whether another neighbor becomes better than the current optimal neighbor due to changes in the network state (e. g., traffic pattern, level of contentions, network topology, etc.). Exploration policy is, also, of high importance at the beginning of network operation, where Q-tables are initialized based on *Shortest Path Routing* and learning agents try to find the best paths based on the current network state. Hence, we adopt an $\varepsilon - greedy$ exploration policy [9] which makes the decision process (the selection of a neighbor to forward an incoming burst) probabilistic, with a small probability ε to select a non optimal neighbor (e. g., $\varepsilon = 0.1$), and a high probability $(1-\varepsilon)$ to select an optimal neighbor (e. g, $(1-\varepsilon) = 0.9$).

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

Convergence is a known issue in reinforcement learning. Indeed, when a reinforcement learning model is applied to solve a problem, it is not always guaranteed that this model will converge to a stable solution. Fortunately, according to [10], a Q-learning algorithm that uses a lookup table representation of estimates is guaranteed to converge, which is the case for RLAR.

To update Q-tables, each time a node forwards a burst to a neighbour, a feedback packet is sent back to the node by the neighbour; this may seem causing considerable overhead. Fortunately, this overhead has almost no effect on the performance of OBS networks: (a) feedback packets are sent using wavelength(s) used only to transmit control traffic; and (b) the size of feedbacks is very small (a numerical value).

Furthermore, no routing is needed since all feedback packets traverse no more than one hop. This is different from existing approaches where an ACK packet is returned back to the source node if the burst is successfully received by the destination node, and a NACK message is returned back to source node if the burst is dropped by an intermediate node (e.g., [4, 5, 7]). These ACK/NACK packets are considerably larger (than control packets in RLAR) since they include routing information from the sending node (destination or an intermediate node) to the source. Thus, the overhead introduced by RLAR is considerably less important than existing approaches.

C. *Loopless routing*

When distributed routing is used, loops may occur in routing paths, i.e., a packet (burst) may pass through the same node more than once. In the worst case, a packet may circulate indefinitely in a loop causing an increase in the network load and, consequently, an increase in the burst loss probability. In order to prevent loops from appearing in RLAR , we modify the loopless routing algorithm proposed in [11] (which aims to find one deflection alternative in each node with respect to a given destination) and use it with RLAR. Our modification is in the step 3 of this algorithm where all of the forwarding alternatives of the selected node are added to the list of forwarding alternatives instead of only one forwarding alternative in the original algorithm (see [11] for more details).

IV. THE INTEGRATED APPROACH

The integrated approach, called Integrated *Reinforcement Learning-based Routing and Contention Resolution* (IRLRCR) approach, aims to adopt both a proactive approach and a reactive approach. This is to (1) attempt to prevent contentions from appearing in the OBS network; and (2) attempt to resolve contentions after they appear. For that, in IRLRCR, RLAR is adopted as the proactive approach and RLDRS [3] is adopted as the reactive approach using deflection routing. This combination is obvious since RLAR and RLDRS adopt the same general approach adopted by Q-learning algorithm with lookup table representation of estimates to take decisions. Indeed, in IRLRCR, we use only one lookup table, called *Global Table*, which is used by both RLAR and RLDRS as the Q-table and the Deflection Table, respectively. In the normal case, bursts are routed using RLAR; however, when a contention occurs and a burst has to be deflected, RLDRS is used. Thus, since both RLAR and RLDRS use the same lookup table, the neighbor with the second highest Q-value is selected by RLDRS to deflect a burst.

Fig. 3 shows the operation of IRLRCR in an OBS core node. We can see that RLAR and RLDRS are used in harmony and with a sequential manner. Nevertheless, because of the $\varepsilon - greedy$ exploration policy used by both RLAR and RLDRS, even it is very small, the possibility that RLAR and RLDRS select the same output port in the same selection process, exists. To tackle this problem, we add a simple test to check if the output port selected by RLAR is the same as the one selected by RLDRS. If it is the case, another output port is selected. The new output port is the one with the highest Q-value other than the current output port (selected by both RLAR and RLDRS).

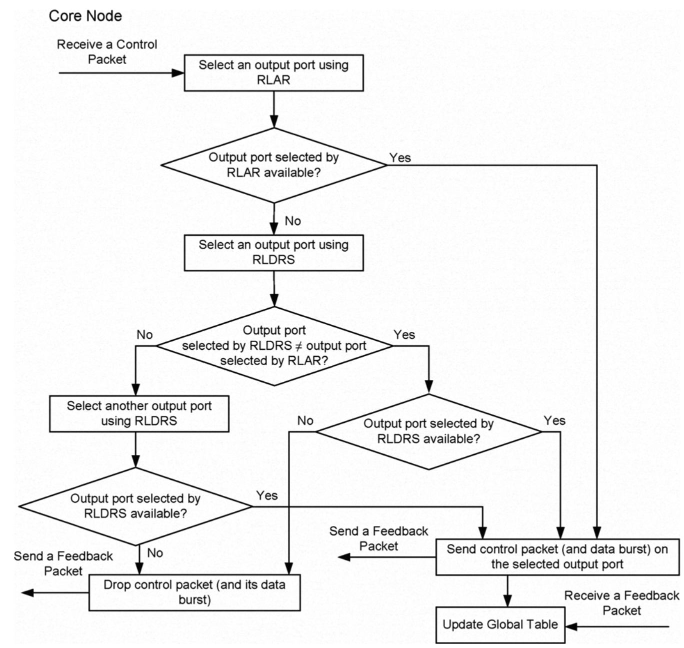


Figure 3. The operation of IRLRCR at an OBS core node.

V. RELATIVE QUALITY OF SERVICE DIFFERENTIATION

IRLRCR and RLAR can provide relative QoS differentiation in the OBS network. Relative QoS differentiation is the ability to provide higher level of QoS to a given class of traffic. This differentiation is, generally, based on a given metric, such as loss probability or end-to-end delay. For example, whereas mission critical applications are loss-sensitive and then require very low loss probability, Voice over Internet Protocol (VoIP) and Video On Demand (VOD) applications are delay-sensitive and then require low end-to-end delay.

Offset-based QoS [12] was the first proposed scheme to provide relative QoS differentiation in OBS networks. In this scheme, high priority bursts are assigned an extra offset time to allow them reserving resources in advance compared to low priority bursts; the scheme offers complete priority isolation but at the cost of additional end-to-end delay. In addition,

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

offset-based QoS scheme favors bursts with shorter lengths [13]. In [13], a proportional QoS scheme is proposed. This scheme keeps track of the proportional loss probability among traffic classes of different priorities; if the proportional differentiation rule is violated, some low priority bursts are intentionally dropped. We note that the intentional burst dropping results in a higher overall loss probability.

In this section, we aim to exploit the ability of IRLRCR in reducing loss probability to provide relative QoS differentiation. This is performed without extra offset time or intentional dropping. In addition, no burst reservation preemption is used (i.e., a higher priority burst cannot preempt the resource reservation of a lower priority burst). We consider both loss-sensitive traffic and delay-sensitive traffic. Hence, we consider two classes of traffic: (a) *classe-1* which is loss sensitive traffic; and (b) *class-2* which is delay-sensitive traffic. Whereas *class-1* traffic has to be routed inside the OBS network with the lowest possible loss probability, *class-2* traffic has to be routed inside the OBS network with the lowest possible end-to-end delay.

We suppose that each OBS edge node has an assembly buffer for each destination and each class of traffic (i.e., *class-1* or *class-2*). Whenever a packet arrives at the OBS edge node, it is classified as *class-1* or *class-2* according to its destination and its QoS requirement. After the burst assembly process: (1) *class-1* bursts are routed, through the OBS network, using IRLRCR which offers the lowest loss probability (see Section VI); and (2) *class-2* bursts are routed using RLAR which offers the lowest end-to-end delay with significant improvement in terms of loss probability compared to Shortest Path routing (SP) (see Section VI). It is worth noting that SP can be used for *class-2* traffic, however, while this choice will decrease slightly end-to-end delay, it will have a negative impact on the loss probability performance of this class of traffic.

Our proposed scheme is able to provide QoS differentiation in simple and efficient manner since: (a) no intentional dropping is applied for low priority bursts which decreases the overall loss probability in the OBS network; (b) no extra offset time is added to the basic offset time which increases the overall end-to-end delay; and (c) low priority burst reservation preemption is not adopted since preemption is complex to implement and could reduce resource utilization in the OBS network [14]. In addition, it is not appropriate to use preemption in this scheme since there are no low priority and high priority classes; each class has its own requirements in terms of loss probability and delay.

VI. SIMULATION RESULTS AND ANALYSIS

In this section, we present simulation results that we have performed to evaluate the performance of RLAR and IRLRCR. We use the ns-2 simulator [15] and modules that implement OBS in ns-2 [16]. We consider two kinds of topologies, namely, mesh topologies represented by NSFNET with 14 nodes and regular topologies represented by regular 4 x 4 nodes torus topology. We assume that each single fiber link is bidirectional and has the same number of wavelengths. Each node in the network can generate, route and receive traffic. Sources and destinations of traffic connections are generated randomly between any two nodes in the network. The traffic load is expressed as the percentage of the total load that can be carried by the network (i.e., Traffic Load = (Offered Load) / (∑ Link capacities)) where Offered Load is the amount of traffic injected, per second, in the network. The capacity of a link is the sum of the capacities of all the wavelengths in this link. We use *Min-Burst length Max Assembly Period* (MBMAP) algorithm for burst assembly [17], with maximum burst size fixed to 10 kB and LAUC-VF (Last Available Unused Channel with Void Filling) algorithm [18] for wavelength assignment in OBS edge nodes. In our simulations, we use exponential ON/OFF traffic unless stated otherwise. All of the following results have a confidence level of 95%.

A. *Results of RLAR*

The goal of these simulations is to measure the performance of RLAR in terms of loss probability which is the main performance metric in buffer-less OBS networks. In addition, since RLAR uses, in general, longer routing paths than *Shortest Path First* routing, we evaluate RLAR in terms of mean burst end-to-end hops. We compare RLAR to both *Shortest Path First (SPF)* routing and *Q-learning algorithm for Path Selection (QPS)* proposed in [4]; the motivations behind this choice are (a) SPF is considered as the standard routing algorithm, not only in OBS networks, but also in most of data communication networks; (b) QPS is the most recent work on multi-path routing in OBS networks (to the best of our knowledge) that outperforms existing approaches (e.g., [5]); and (c) QPS represents the first attempt to use reinforcement learning in the routing of OBS networks [4].

Initially we set (a) the learning rate α to 0.6 (other values of α have a slight impact on the trend of the simulation results; they will not be presented because of space limitation); (b) the period to measure the loss rate on each link (B_{xy}) to 2 seconds; and (c) the exploration probability ε to 0.02. For QPS parameters, we use the same values reported in [4].

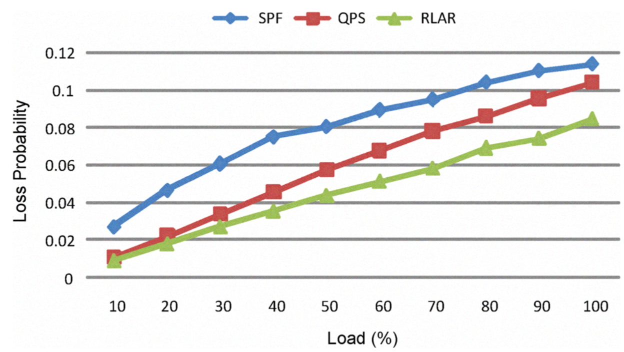
Fig. 4 (a) shows burst loss probability when varying the offered load on NSFNET. We can see clearly that RLAR outperforms both QPS and SPF. Indeed, The relative improvement (defined as [(Loss with SPF (QPS) – Loss with RLAR) / (Loss with SPF (QPS))]) of RLAR compared to SPF is about 66% at load 10% and about 25% at load 100% and compared to QPS it is about 15% at load 10% and about 18% at load 100%. Fig. 4 (b) shows that SPF has the lowest mean end-to-end delay (in this paper the end-to-end delay represents the hop distance between the source node and the destination node), which is expected. It also shows that RLAR improves significantly end-to-end delay compared to QPS. Indeed, whereas mean burst end-to-end delay is around 2.6 for QPS, it is around 2.2 for RLAR and 2.1 for SPF. This improvement is particularly significant in backbone networks where link propagation delay is typically measured in milliseconds.

Fig. 5 (a) shows the loss probability when varying the offered load in the regular 4 x 4 topology. The improvement of RLAR is better using this topology compared to NSFNET; at load 100% the relative improvement of RLAR when compared to SPF is about 65% and compared to QPS is 41%. This can be explained by the fact that in this topology, the average node

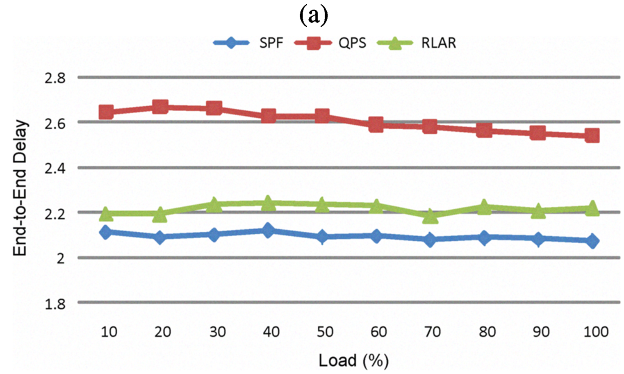
Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

degree is 4, whereas the average node degree in NSFNET is 3. This supports the idea that RLAR performs better whenever the average node degree increases, due to the increase in the number of forwarding alternatives in each node in the OBS network. Fig. 5 (b) shows mean burst end-to-end delay with regular 4 x 4 topology. Here again, RLAR outperforms QPS and slightly underperforms SPF.

Fig. 6 (a) shows burst loss probability when fixing the traffic load at 60% and varying the number of wavelengths from 8 to 128 wavelengths. We can see clearly that regardless of the number of wavelengths, RLAR outperforms SPF and QPS. Fig. 6 (b) shows burst end-to-end delay when fixing the traffic load at 60% and varying the number of wavelengths from 8 to 128 wavelengths. Here again, RLAR outperforms QPS regardless of the number of wavelengths.

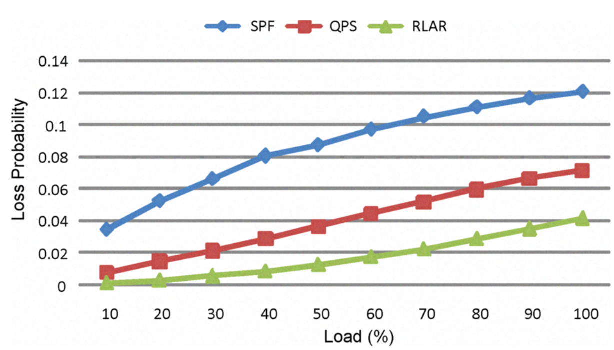


(a)

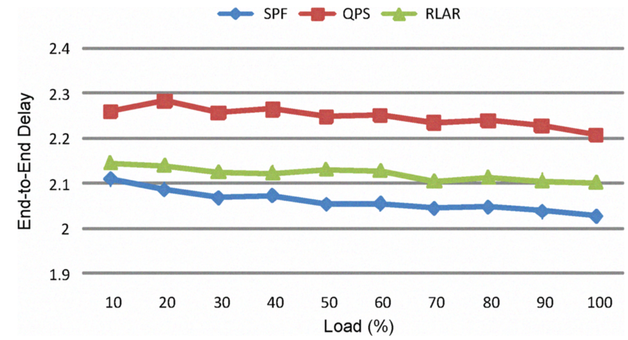


(b)

Figure 4. RLAR: Loss and delay vs. Load on NSFNET with 64 wavelengths.

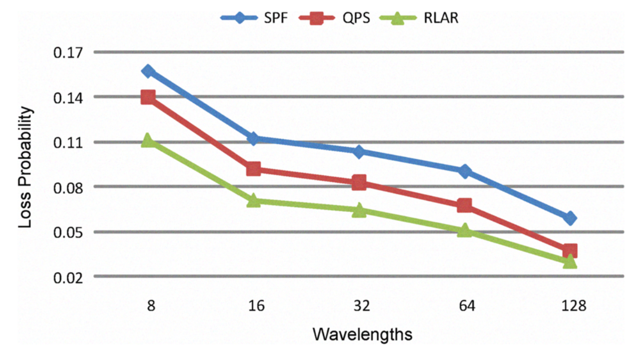


(a)

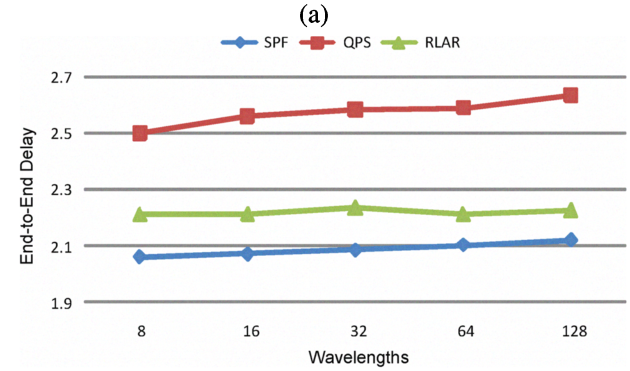


(b)

Figure 5. RLAR: Loss and delay vs. Load on Regular 4 x 4 Topology with 64 wavelengths.



(a)



(b)

Figure 6. RLAR: Loss and delay vs. Number of Wavelengths on NSFNET.

B. Results of IRLRCR

In this section, we measure the performance of IRLRCR. In addition, we compare the performance of IRLRCR to the performance of RLAR and the performance of RLDRS.

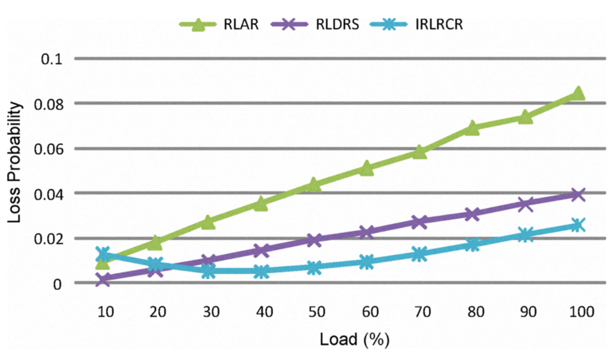
Fig. 7 (a) shows burst loss probability when varying the offered load on NSFNET. We observe here that IRLRCR, clearly outperforms RLAR and RLDRS. Indeed, at load 100%, the relative improvements of IRLRCR over RLAR and RLDRS are about 69% and 35%, respectively. Fig. 7 (b) shows that RLAR has, generally, the lowest mean end-to-end delay and that IRLRCR has the highest mean end-to-end delay. Indeed, the mean end-to-end delay of RLAR, RLDRS and IRLRCR are around 2.2 hops, 2.25 hops and 2.7 hops, respectively. This additional delay of IRLRCR is expected since it uses both RLAR and RLDRS which add the additional delay of each of them to the delay of IRLRCR. However, this delay is acceptable if we consider the considerable improvement of

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

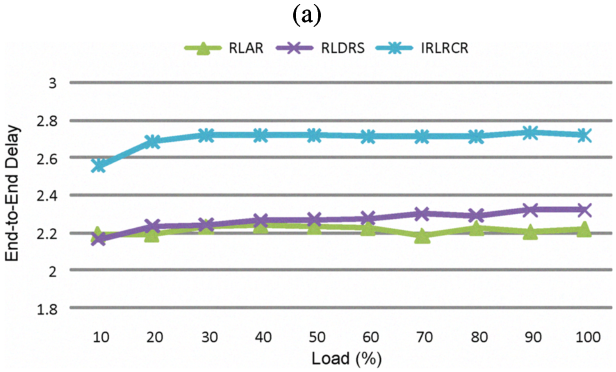
IRLRCR over both RLAR and RLDRS in terms of loss probability.

Fig. 8 (a) shows the loss probability when varying the offered load in the regular 4x4 topology. We observe here that, differently from NSFNET topology, IRLRCR always outperforms both RLAR and RLDRS. Indeed, the mean relative improvements of IRLRCR over RLAR and RLDRS are about 80% and 86%, respectively. Another interesting observation here is that, overall, RLAR outperforms RLDRS in terms of loss probability. Those behaviors (namely, the fact that IRLRCR outperforms both RLAR and RLDRS over all of the loads and the fact that RLAR outperforms RLDRS over all of the loads) can be explained by the fact that the performance of RLAR becomes better when node degrees increase (we recall here that the mean node degree of NSFNET is 3 while the mean node degree of the regular 4 x 4 topology is 4). This confirms our above conclusion that states that whenever the mean node degree increases, multi-path routing becomes more and more efficient in reducing loss probability. Fig. 8 (b) shows mean burst end-to-end delay with regular 4 x 4 topology. Here again, IRLRCR underperforms RLAR and RLDRS for the same reason as in Fig. 7.

Fig. 9 (a) shows burst loss probability when fixing the traffic load at 60% and varying the number of wavelengths from 8 to 128 wavelengths on NSFNET. We can see clearly that regardless of the number of wavelengths, IRLRCR outperforms both RLAR and RLDRS. Fig. 9 (b) shows burst end-to-end delay when fixing the traffic load at 60% and varying the number of wavelengths from 8 to 128 wavelengths on NSFNET. Here again, we see that IRLRCR reduces effectively loss probability at the cost of an increase in end-to-end delay regardless of the number of wavelengths.

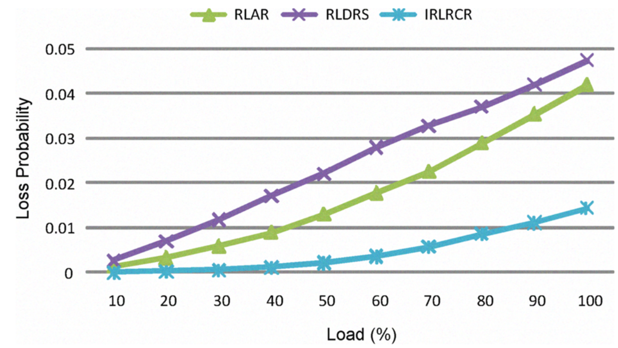


(a)

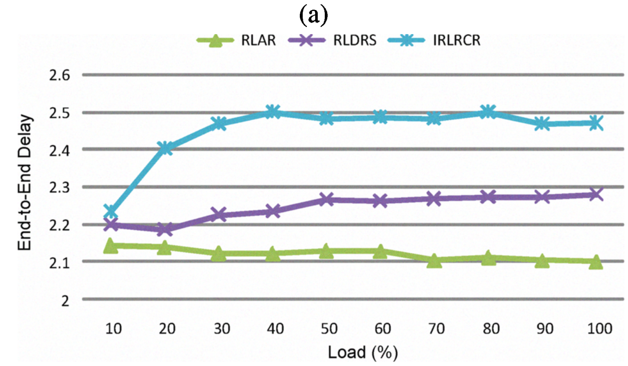


(b)

Figure 7. IRLRCR: Loss and delay vs. Load on NSFNET with 64 wavelengths.

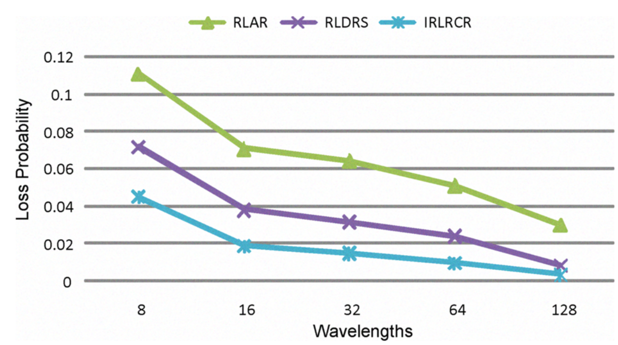


(a)

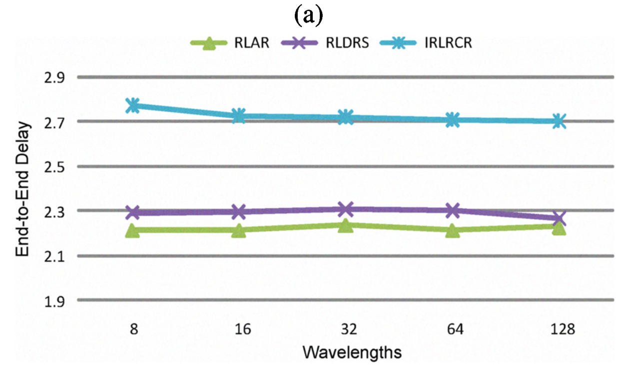


(b)

Figure 8. IRLRCR: Loss and delay vs. Load on Regular 4 x 4 Topology with 64 wavelengths.



(a)



(b)

Figure 9. IRLRCR: Loss and delay vs. Number of Wavelengths on NSFNET.

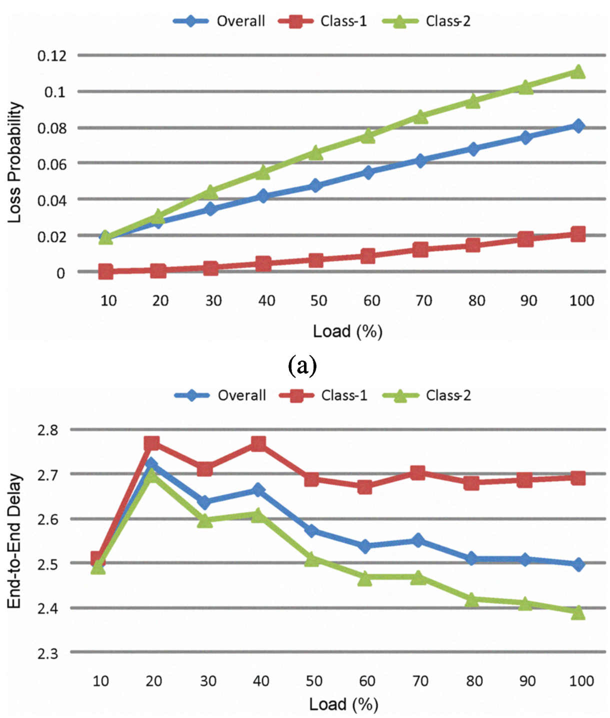
C. Results of QoS differentiation

The goal of these simulations is to measure the capacity of IRLRCR and RLAR to perform QoS differentiation in terms of loss probability and end-to-end delay. As explained in Section V, we consider two classes of traffic: (a) *class-1*: loss-sensitive

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233

traffic; it uses IRLRCR; and (b) *class-2*: delay-sensitive traffic; it uses RLAR. In our simulations, we set the proportion of *class-1* traffic to 33% of the overall traffic, while *class-2* traffic constitutes the rest (other values will have a slight impact on the presented results).

Fig. 10 (a) shows loss probability of *class-1* traffic, *class-2* traffic and the overall loss probability in the network. We can see clearly that *class-1* traffic and *class-2* traffic are differentiated in terms of loss probability. Indeed, whereas the mean loss probability of *class-1* traffic (over all of the loads) is about 0.008, the loss probability of *class-2* traffic is about 0.06 which is roughly eight times the loss probability of *class-1* traffic. We can also observe that the loss probability of *class-1* traffic is in the order of 10^{-4} at load 10% and 10^{-2} at load 100%, which is a remarkably low level of losses for OBS networks without wavelength converters and FDLs. Fig. 10 (b) shows that *class-1* traffic and *class-2* traffic are differentiated in terms of end-to-end delay since *class-2* traffic has clearly lower end-to-end delay compared to *class-1* traffic. The same tendency is observed using regular topology (related figures are not presented because of the space limitation).



(b)

Figure 10. QoS differentiation: Loss and delay vs. Load on NSFNET with 64 wavelengths

VII. Conclusion and Future Work

In this paper, we proposed an adaptive alternative reinforcement learning-based routing approach (RLAR) that reduces effectively burst loss probability compared to Shortest Path routing (SPF) and to a recent and efficient multi-path routing algorithm (QPS). By combining RLAR and the reactive deflection routing approach RLDRS [3], we defined an Integrated Reinforcement Learning-based Routing and Contention Resolution approach (IRLRCR). Simulation results show that RLAR, not only improves burst loss probability but also keeps burst end-to-end delay close to burst end-to-end delay of SPF, and considerably better than QPS. In addition, the performance evaluation of IRLRCR shows that the combination of RLAR and RLDRS is successful since it reduces, effectively, loss probability compared to both RLAR and RLDRS, at the cost of a slight increase in mean end-to-end delay. We notice that the loss probability level of IRLRCR (in the order of 10^{-2} at load 100%) is remarkable for the OBS networks under study (without wavelength converters and FDLs). We also did show the ability of RLAR and IRLRCR to provide QoS differentiation in terms of both loss probability and end-to-end delay. In the future, we plan to propose admission control mechanisms to guarantee specific loss probability thresholds.

References

[1] C. Qiao and M. Yoo, "Optical burst switching (OBS) - a new paradigm for an optical Internet". *Journal of High Speed Networks*, 1999, 8(1), p. 69-84.

[2] C. Qiao and M. Yoo, "Choices, features and issues in optical burst switching". *Optical Network Magazine*, 2000, 1(2), p. 36-44.

[3] A. Belbekkouche, A. Hafid, and M. Gendreau, "A Reinforcement Learning-based Deflection Routing Scheme for buffer-less OBS Networks". in *Proceedings of IEEE GLOBECOM*, 2008, New Orleans, USA.

[4] Y. V. Kiran, T. Venkatesh, and C.S.R. Murthy, "A Reinforcement Learning Framework for Path Selection and Wavelength Selection in Optical Burst Switched Networks". *IEEE Journal on Selected Areas in Communications*, 2007, 25(9), p. 18-26.

[5] L. Yang and G. N. Rouskas, "Path Switching in Optical Burst Switched Networks". in *Proceedings of Networking*, 2005.

[6] G. R. V. Thodime, V. M. Vokkarane, and J.P. Jue, "Dynamic congestion-based load balanced routing in optical burst-switched networks". in *Proceedings of IEEE GLOBECOM*, 2003.

[7] D. Ishii, N. Yamanaka, and I. Sasase, "Self-learning Route Selection Scheme using Multipath Searching Packets in OBS Network". *Journal of Optical Networking*, 2005, 4(7), p. 432-445.

[8] S. Ganguly, S. Bhatnagar, R. Izmailov, and C. Qiao, "Multi-path adaptive optical burst forwarding", in *Workshop on High Performance Switching and Routing, HPSR* 2004.

[9] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction", 1998, Cambridge, Massachusetts: The MIT Press.

[10] P. D. Christopher and J. C. H. Watkins, "Q-Learning". *Machine Learning*, 1992, 8, p. 279-292.

[11] J. P. Jue, "An algorithm for loopless deflection in photonic packet-switched networks". in *Proceedings of IEEE ICC*, 2002.

[12] M. Yoo, C. Qiao, and S. Dixit, "QoS performance of optical burst switching in IP-over-WDM networks". *IEEE Journal on Selected Areas in Communications*, 2000, 18(10), p. 2062-2071.

[13] Y. Chen, M. Hamdi, and D.H.K. Tsang, "Proportional QoS over OBS networks". in *Proceedings of IEEE GLOBECOM*, 2001.

[14] A. Maach, A. Hafid, and A. Belbekkouche, "Anticipated Retransmission-based Contention Reduction Schemes in Optical Burst Switching Networks". in *Proceedings of SCS/IEEE SPECTS*, 2008.

[15] "NS-Simulator", www.isi.edu/nsnam/ns.

[16] S. Gowda, R. K. Shenai, K. M. Sivalingam, and H. C. Cankaya, "Performance evaluation of TCP over optical burst-switched (OBS) WDM networks". in *Proceedings of IEEE ICC*, 2003.

[17] X. Cao, J. Li, Y. Chen, and C. Qiao, "Assembling TCP/IP packets in optical burst switched networks". in *Proceedings of IEEE GLOBECOM*, 2002, Taipei.

[18] Y. Xiong, M. Vanderhoute, and H. C. Cankaya, "Control architecture in optical burst-switched WDM networks". *IEEE Journal on Selected Areas in Communications*, 2000, 18(10), p. 1838-1851.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7233
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7233