Distributed Resources Assignment for Optical Burst Switching without Wavelength Conversion (Invited Paper)

Cristina Cervelló-Pastor, Joan Triay, and Sebastià Sallent
Department of Telematics Engineering
Universitat Politècnica de Catalunya (UPC)
Esteve Terradas 7, 08860 Castelldefels, Spain
Emails: {cristina, joan.triay, sallent}@entel.upc.edu

***Abstract*—The amount of bursty Internet traffic leads to develop new architectures and technologies, such as Optical Burst Switching (OBS), to efficiently satisfy future bandwidth requirements. Burst loss probability is an important quality of service metric for OBS due to its bufferless characteristic, even more critical without wavelengths converters. So, resource assignment is an important issue to solve in OBS networks. In this paper, two distributed resources assignment schemes without wavelength conversion capability are proposed. Whereas the first one is applied at the edge nodes to achieve a loss-free core network, the second is an enhanced routing and wavelength assignment scheme applied at core nodes. Simulation results indicate that the first scheme offers a loss-free solution with blocking probability only at ingress nodes and high traffic load. The second one reduces the network-wide burst loss probability significantly compared with other schemes.**

I. Introduction

Widespread deployment of triple-play services (data, voice and video) leads to a growing amount of bursty Internet traffic. This, jointly with the rapid advance of optical technologies, has driven the development of new architectures to provide the flexible and dynamic resources allocation to support this traffic. Optical burst switching (OBS) has emerged as an efficient optical switching technology for satisfying the future bandwidth requirements of the next generation optical Internet. OBS avoids the inefficient resource utilization of optical circuit switching (OCS) and the requirements of optical packet switching (OPS) in terms of optical buffers and processors.

A fundamental issue to deploy OBS networks is that of resolving the burst contention problem in order to reduce the burst loss probability. Since optical burst-switched networks provide a connectionless transport, there exists the possibility that bursts can contend at intermediate nodes. Contention occurs if multiple bursts from different input ports try to leave the OBS switch through the same output port at the same time and on the same wavelength. In view of this, effective contention resolution is critical in OBS networks to restrict losses to a reasonable low level.

In this paper we give an insight into two different resources assignment schemes. The first one works as a wavelength assignment scheme jointly with a medium access controller which delivers zero burst losses under low-to-medium network loads, and at high loads these only happen in the ingress buffers at the edge nodes of the OBS network. The second one is a distributed routing and wavelength assignment scheme which enhances the resources assignment while acting against congestion stages through the use of ACK control packets.

The remainder of the paper is as follows. Section II gives a short brief about resources assignment for OBS networks. In the next two sections, III and IV, we introduce the two proposed schemes. Results through simulations are analyzed in section V, and finally section VI presents the conclusions of the paper.

II. Resources Assignment for OBS

Resources assignment plays a key role in OBS networks, specially under certain constraints, such as wavelength continuity. Thus, the better we manage the resources, the more likely to enhance the performance of the network. Resources assignment comprises many factors, processes and issues and as such, it can be implemented on several parts of the network. To name a few, routing and wavelength assignment involves the search for the best routes and wavelengths to carry out both the transmission and switching of a burst. In burst scheduling processes, the objective is to improve the utilization of the optical channel while minimizing the number of burst blockings due to the contention for the use of the same resource. Contentions may occur among bursts that use the same wavelength and route that either collide or overlap [2]. In this scenario, we take into account both, collisions and overlaps, to define the resources assignment mechanism.

In this sense, and in order to provide a comprehensive description of the protocol architectures introduced in the paper, we will classify the resources assignment schemes into two big groups. In the first group we fit the schemes that can be applied mainly at the edge of the network, whereas in the second one, those that are more likely in the core.

The architectures that we present next are characterized by being totally distributed, thus, in neither of the two proposals there is a central node deciding about the assignment of resources.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792

III. Distributed Resources Assignment at the Edge of the Network

Many different resources assignment techniques can be applied at the edge of the network, that is, on the ingress nodes. Nevertheless, the access control to the optical channel is maybe the easiest to implement. Medium Access Control (MAC) protocols provide a means to control the transmission into the channel depending on the state of the network, becoming a very interesting solution to provide loss-free schemes under specific network conditions. This first proposal deals with a distributed MAC protocol for optical burst switching. One of the main advantages of Distributed Access for OBS (DAOBS) is that guarantees zero losses for transit bursts, that is, in the core of the network, as bursts are only transmitted when there exist free resources. However, due to the distributed operation of the protocol and the control packet propagation, a slight extra delay is added that, in most situations, it does not represent a great issue in comparison with other schemes taking into account its loss-free property for switched bursts.

Although DAOBS is similar to a light-trail [3], the first also provides a burst MAC with fairness among nodes belonging to the optical bus. Likewise, DAOBS can be extended onto more complex topologies, like light-mesh ones. Nevertheless, the later needs electronic buffers to ensure that the burst/packets are going to be delivered, whereas in our case, DAOBS ensures the delivery of a transmitted burst neither using buffers nor wavelength converters.

A. Basic operation of DAOBS

In this section we briefly describe the original operation of DAOBS. Further information can be found in [4]. Later, we will illustrate some of the enhancements required to use the protocol on a mesh topology. For better comprehension, we will first exemplify its use along a light-path or optical bus (see Fig. 1a).

The basic piece of network that can manage the distributed protocol is a bus or light-path between two nodes. In such a case, we consider two unidirectional control channels which can be in-fiber (i.e. using a specific wavelength): the downstream or forward channel which forwards Burst Control Packets (BCP) and goes from the Head of the Bus (HoB) node to the Tail of the Bus (ToB) node, and the upstream or reverse channel, on which Request Control Packets (RCP) are forwarded from the ToB to the HoB. The bus is normally composed by several extra nodes between the HoB and ToB. All these nodes, except the ToB, can access and transmit their bursts on this bus according to the operation of the DAOBS protocol. BCPs have a BUSY bit for announcing the occupation of the upcoming slot, whereas RCPs have a REQ bit for letting know the rest of upstream nodes the request for a free slot.

In the proposed architecture, a DAOBS entity in a network node is identified by its *input port* + *output port* + *wavelength*, and a DAOBS bus by its *HoB* + *ToB* + *wavelength* on the whole network. Therefore, a node can have multiple entities, which are related to the number of wavelengths it can use. We should note that the number of wavelengths accessible at a node depends on the burst instantiation method. On a wavelength continuity constraint WDM network, this instantiation can be done by running a graph coloring algorithm or using a standard optical reservation scheme such as GMPLS.

A DAOBS entity in a node is composed by (see Fig. 1a):

- Two counters: a request counter (RQ_C) and a countdown counter (CD_C).
- A local queue (LQ).
- A distributed queue (DQ).

Figure 1a depicts the relationship between these elements. The LQ queues bursts waiting to access the optical channel, whereas the DQ is a one-position queue that stores the next burst to transmit on the bus. The so-called virtual distributed queue of a bus is then made up from the collection of DQs, one from each node that belongs to the bus.

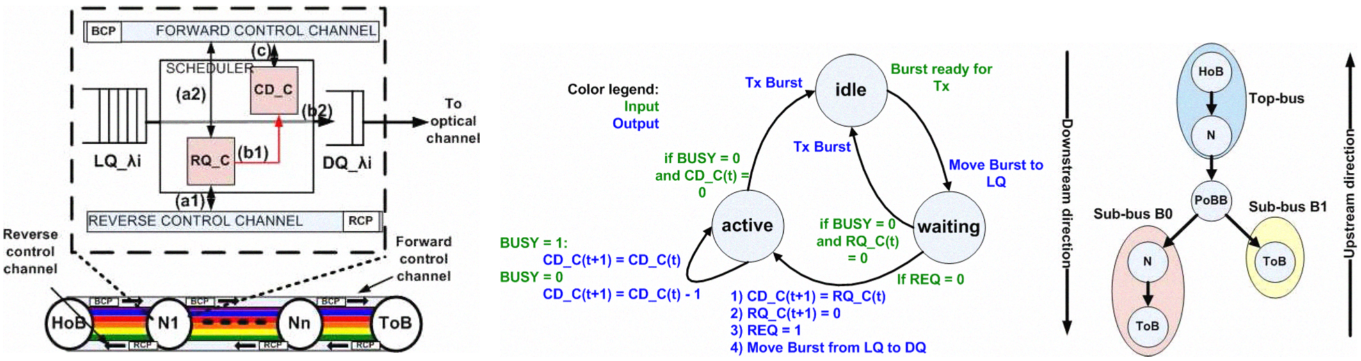
The protocol operates as follows. Each node can be in any of the following three states, as shown in Fig. 1b: *idle*, in case the node has nothing to transmit on the corresponding optical bus; *waiting*, meaning that it has something to transmit but it has not made a successful request yet; or *active*, in case that the node has made a request and is waiting a free slot to transmit the corresponding burst.

The LQ receives bursts from the Virtual Output Queues (VOQ) after the wavelength assignment module has successfully processed them. Initially, all the entity counters are reset to zero. While a node is *idle*, it monitors the RCPs on the reverse control channel and BCPs on the forward control channel, and increments (see Fig. 1a action (a1)) or decrements (a2) the RQ_C for every REQ = 1 in the RCP, and for every BUSY = 0 in the BCP, respectively. As soon as a burst enters the LQ, the node changes its state to *waiting*. In this state, the node continues monitoring the BCPs and RCPs. Depending on the state of the counters, the node can become *active* using a free RCP (REQ = 0), or a free BCP (BUSY = 0) in the specific case that any of the downstream nodes has previously made a request (RQ_C = 0). In the second case, the node does not need to make a request and can occupy the upcoming slot as soon as it starts. This enhancement minimizes the average delay to access the channel specially under low-to-medium traffic loads. During the activation, the scheduler dumps the value of the RQ_C to the CD_C (b1) and moves the first burst from the LQ to the DQ (b2). Next, the node monitors the control packets on the forward channel to decrement the CD_C (c) for each free forwarded BCP (BUSY = 0). When this counter reaches 0, the node can use the upcoming free slot and transmit the burst stored in the DQ.

B. Distributed Access on Mesh Topologies

Hereafter we describe the changes required to adapt the distributed access protocol to work on fully or partially meshed network topologies. The basic piece of network to enable the use of the DAOBS in the original protocol [4] was the optical bus, easily adaptable to a ring. Now, the original protocol architecture is spread over more complex networks by means of tree-based wavelength topologies.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792



(a)

(b)

(c)

Fig. 1. Distributed Access for OBS: (a) DAOBS entity. (b) General operation flow. (c) DAOBS tree topology.

In OBS without wavelength conversion we have to face two different kinds of losses [2]. On the one hand, there exist collisions when from two different input ports we need to switch two or more bursts through the same output port on the same wavelength. On the other hand, we can also have collisions when multiplexing the transmission of bursts on a certain node. In this case, a node may reserve resources to transmit a burst through an output port on a given wavelength without knowing that an upcoming one needs to reserve the same resources at the same time, which will produce a collision as a result of an overlap. The first type of collision can be solved using a tree-based topology of wavelengths which guarantees that a network node cannot receive from two input ports a burst to the same output port, thus avoiding burst switching collisions. The second type is more difficult to be solved without using other contention resolution schemes. In this case, DAOBS is specially useful as it gives priority to upcoming switched bursts and only permits the transmission of the own ones when there exist free resources on the optical channel, according to the monitoring process of the RQ_C and CD_C counters.

The tree-based topology operates like a multiple bus topology rooted at the topmost node (see Fig. 1c). This root node is in charge of generating at every time slot the BCP announcing the occupation of a slot. Besides, nodes that are point of branch have extra capabilities to determine the proper forwarding of the control and request messages. This class of nodes isolate the different subbranches of a tree, that is, they operate as a barrier between child and parent buses, so that requests from child buses are not propagated to the top-bus. In this case, they keep counting the requests done by their downstream nodes and deliver free/occupied slots according to this number and the occupancy of the upcoming slot from the parent bus. Taking into account this, in the new mesh architecture we define a new type of node which complements the other three already present in the original DAOBS protocol. Henceforth, we classify the nodes as:

- Head of Bus (HoB): It is the topmost node of the tree, the root node.
- Tail of Bus (ToB): It the lowest node on the tree, or in other words, a leaf node.
- Point of Bus Branching (PoBB): A node at which the tree is branched into two or more new branches (child buses).
- The rest of nodes: Nodes that are not any of the previous types.

C. Wavelength assignment

The wavelength assignment is responsible for assigning bursts to a specific bus. Each wavelength on every output port can only be instantiated by a concrete bus, hence assignment either to a bus or a wavelength means the same.

We run an algorithm to carry out this wavelength assignment according to the values of the counters RQ_C and CD_C. As we have previously introduced, the value of these counters determines the position of the node in the distributed FIFO queue. Depending on these values, the node will take more or less time to transmit a burst on that specific bus, thus incrementing or decreasing the channel access delay, and consequently, the end-to-end delay of the burst. In this process, the bus that is expected to provide the lowest access delay is selected.

IV. Distributed Resources Assignment in the Core

Regarding resources assignment in the core of the network, that is, for switched bursts, there exist also a number of techniques. If we assume that all-optical wavelength conversion is available at each node, scheduling protocols are quite important. Their objective is to find a void space on an output wavelength and port. However, without wavelength conversion, we must run other solutions to enhance the assignment of resources or even to resolve burst contentions. In the last case, deflection routing is a reactive space-based domain contention resolution scheme that solves the contention sending the contending burst to a different output port of the switch and, consequently, on a different route towards the destination. Nevertheless, this scheme provides only limited performance improvements. This type of solution is mainly used as a method to reactively reduce the number of contentions without taking into account a planned behavior about the state of the node and the availability of present resources.

In this section we introduce a protocol [5] based on the Ant Colony System (ACS) algorithm by Dorigo et al. [6] which

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792

can be used to provide a more efficient resources assignment on OBS networks without wavelength conversion. One of the advantages of the protocol is that it is totally distributed and therefore, routing or wavelength assignment decisions are taken using only the local information that the network node has at the moment. Using this protocol, the end-to-end path from the origin OBS node to the destination one is not known a priori, which means that the next node towards the destination is calculated at every hop. In order to make independent the offset value from the number of hops along the final traveled burst route, we chose to implement an offset time-emulated approach [7], so that offset can remain constant during the burst forwarding.

In a similar way to the behavior of an ant colony in their task of foraging for food, with forward and backward ants, in the present architecture we define two types of control packets: the Burst Control Packet (BCP) and the Acknowledgement BCP (ACK-BCP). The first is used to signal the reservation of resources along the path traveled by the burst, whereas the second is used to acknowledge the positive or negative delivery of the burst at its destination. The provisioning scheme uses one-way reservation and therefore, ACK-BCPs do not confirm the resources reservation to proceed with the burst transmission.

In the protocol, wavelength selection uses the same burst switching/routing tables. Thus, the actions taken using the routing rules have a direct impact on the wavelength assignment (RWA). The RWA algorithm chooses the lambdas and output ports following the state transition rule, but taking into account that the input port is the node itself. We can emphasize the use of shorter paths in the initial assignment process assigning different values in the RWA rules.

Finally, the protocol is highly customizable and almost every rule/function used in the algorithm is bound to a parameter. Depending on their value, the actions to be taken in the burst routing/switching can be reinforced or neglected. As a result, the performance of the protocol can be highly dynamic which makes it very suitable for changing network environments. However, as a counterpart, if this parametrization is done wrongly, the performance can decrease, and therefore, care must be taken when establishing its values.

A. Resources information

In the protocol, the switching and routing tables store information about the suitability to make a certain transition. Following the analogy with the ant colony example, these values are the pheromone concentrations. These are noted by $\tau_{i,j,k}$, and are computed according to two kinds of information. The first type is the congestion level, which is a measure of the number of contentions through a specific output port and wavelength. The higher the number of contentions, the lower the value should be. And the second is the path length. The shorter the path, the greater the value through a given output port shall be, hence we favor paths that imply less consumption of optical resources. Regarding the sub indexes of τ, i is the input port, j represents the output port, and k is the switched wavelength.

In comparison to other ant-based algorithms for OPS/OBS networks [8] [9], in our case, the number of free wavelengths is no longer a good measure for the congestion level due to the absence of wavelength converters in our network, i.e. following the wavelength continuity constraint.

In the election of the output port, ACS also provides an extra value, called desirability and represented by η. This value gives heuristic information about the attractiveness or desirability of a certain move, for example, in case where the mean distance (or cost) of paths shall be minimized. Thus, the desirability is defined as $\eta_{nj} = \frac{1}{f(x)}$, where $f(x)$ is equal to the length of the shortest path from node n, through output port j to destination m, as defined by $f(x) = |\, x_{n,j}^{m+}(t) \,|$.

B. Processing rules in RWA

The algorithm runs three rules or functions to decide on the routing of the bursts and the update of the tables values used for such purpose. In [5], a detail description and formulae of each of these rules is given.

The first one, the *state transition rule*, chooses the next hop in the routing of the BCP (i.e. the burst) towards its destination. The aim is to select the best j output link/port taking into account that the burst is going to be switched from the i input port on wavelength k. The chosen transition rule is a pseudo-random-proportional action rule that explicitly balances the exploration and exploitation abilities of the algorithm to look for a suitable output port. In this function, the control parameters are β and r_0. The greater the value of β, the more importance is given to choosing an output port belonging to the shortest path. The second parameter is r_0, which can take values in between $r_0 \in [0, 1]$. When its value approaches 1, exploration is neglected.

The *local updating rule* is used to diversify the routing performed by consecutive bursts from different input ports over the same wavelength and probabilistically restrict or favor the use of certain output links. The local updating rule is only applied after a successful switching reservation and before forwarding the BCP (burst) to the next node towards destination. Again, in this case, there is a parameter, α, for controlling the application of the rule. When its value is 0, the function is not applied, and the state information store in the tables during the switching of the burst remains the same.

The last function is the *global updating rule* and it is run by the returning ACK-BCP along the reverse path used by its corresponding burst. The aim of the rule is to reinforce or weaken the values of the switching tables of each node along the route. The new value is calculated assuming a cycle implementation, in which the values are subject to exponential decay, that is, they decrease at a rate proportional to a given value. In our case, this value depends on the length of the path followed by the burst. Thus, if the route followed to reach the destination is the shortest (or equivalent) and the reception of the burst is correct, the added value is maximum. The parameter used for controlling the update of the switching

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792

```
variables: $W$ wavelengths, $N$ nodes;
Initialize parameters $\alpha_1$,$\alpha_2$,$\rho_1$,$\beta$,$\omega$,$\phi$,$r_0$;
for each node $n \in N$ do
   Initialize routing tables;
   $m \leftarrow N - \{n\}$;
   for each possible destination $m$ do
      Initialize candidate nodes list $\mathcal{N}_n^m$;
      Compute initial $\eta_{nj}$ using $\mathcal{N}_n^m$;
      Initialize candidate lambdas list $\mathcal{W}_n^m \leftarrow W$;
   end for
end for
loop
   if burst to transmit then
      create BCP and run ACRWA algorithm;
   end if
end loop
```

Fig. 2. Initialization algorithm.

table is again a user-specified value ρ_1. If it is small, then the values are increased/decreased slowly, and if it is greater, the previous experience, i.e. old values, is neglected in favor of more recent experiences, i.e. increments/decrements triggered by recent ACK-BCPs.

C. The Ant Colony RWA (ACRWA) algorithm

The ACRWA protocol is composed of two main algorithms. The algorithm shown in Fig. 2 is responsible for running the first stages such as the initialization of the parameters values and routing tables. A supporting routing protocol is necessary to get the initial list of routes. In this case, the routing table is filled with the k-shortest routes to every other node of the network using on each route a different output port. Using these routing tables, each node, n, builds the candidates list $\mathcal{N}_n^m$ of neighbor nodes to get to destination m. The tables are also used to calculate the desirability values η_{nj} of $\mathcal{N}_n^m$. Likewise, the list of wavelength candidates, $\mathcal{W}_n^m$, is initialized, which for simplicity is set to W, the total number of wavelengths available on the network. When all these steps are completed, the nodes are ready to process burst transmission queries.

Next algorithm in Fig. 3 shows the specific behavior followed by the BCP (burst) on its way to the destination node. Initially, the origin node runs the RWA process to select the output port and wavelength over which the burst is going to be transmitted. After this initial stage, the burst is transmitted and routed to its destination. At every hop, the BCP runs the transition rule. The first step in this process, is the computation of a random number which is used to decide if the burst is going to exploit previous knowledge or try to explore for new possible routes. If the node exploits, it checks the list of candidates to route the burst and calculates the next node using the state transition rule. It may happen that using the selected output port, there occurs a collision. In this case, and assuming the absence of a burst contention resolution scheme, the burst is blocked and the global updating rule is initiated using a negative feedback on the reverse path, i.e. an ACK-BCP with a negative delivery of burst. Otherwise, if the burst finally arrives to destination, it means that the transmission has been successful. This triggers the start of the global updating rule, but now using a positive feedback. The algorithm finishes when the feedback ACK-BCP arrives the origin of the burst. In the algorithm, we use the notation (t) to remark the possibility that some values can change throughout time.

```
$x^m(t) \leftarrow \oslash$;
$Burst_{blocked} \leftarrow false$;
if $x^m(t) = \oslash$ then
   run initial RWA;
end if
repeat
   if $\exists u \in \mathcal{N}_n^m(t)$ then
      $r \leftarrow random()$;
      if $r \leq r_0$ then
         for all $u \in \mathcal{N}_n^m(t)$ do
            Choose $j \in \mathcal{N}_n^m(t)$;
         end for
      else
         for all $u \in \mathcal{N}_n^m(t)$ do
            Calculate empirical prob. dist. $\mathcal{F}_u{}^m(x)$;
         end for
         Choose $j \leftarrow \mathcal{F}_u{}^m(x)$;
      end if
      $x^m(t) \leftarrow x^m(t) \cup \{link(n,j)\}$;
      if burst reservation $link(n,j)$ is false then
         $Burst_{blocked} \leftarrow true$;
      end if
      if $Burst_{blocked}$ is false then
         for all $u \in \mathcal{N}_n^m(t)$ do
            if $link(n,u) = link(n,j)$ then
               run positive local updating rule;
            else
               run negative local updating rule;
            end if
         end for
      end if
   else
      $Burst_{blocked} \leftarrow true$;
   end if
until (burst arrives destination) OR ($Burst_{blocked}$ is true);
repeat
   $x^m(t) \leftarrow x^m(t) - \{link(n,j)\}$;
   run global updating rule;
until $x^m(t) = \oslash$ OR ACK-BCP arrives origin node;
```

Fig. 3. ACRWA algorithm.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792

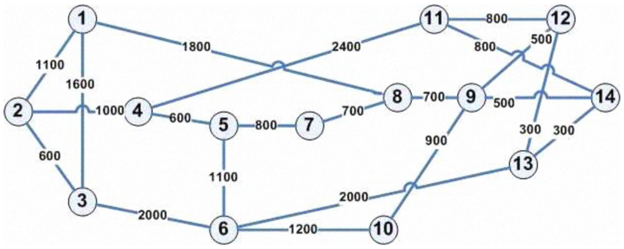


Fig. 4. NSFNET network.

Regarding the complexity of the ACRWA algorithm, consider the network modeled by an undirected graph $G = (V, E, W)$ where V represents the set of vertices (nodes), and E represents the set of edges (links) each one with W wavelengths. The worst case algorithm complexity added by the forward and backward BCP of a burst throughout its traveling period will be $O(VW + V^2) \sim O(V^2)$, if $V \sim W$, calculated as follows: the initial wavelength assignment, which implies the lookup of the best wavelength and output port to transmit the burst, adds a cost of $O(VW)$. Once the lambda is assigned, the computational complexity added at every hop of the BCP is $O(V)$, taking into account that the transition and updating rules imply an operation that can be run in constant time ($O(1)$), but must be applied at every possible node in the candidates list, which at the most can be in the order $O(V)$. As a result, the forward and backward processing of the BCP and ACK-BCP adds a complexity of $O(V^2)$. It is worth noting that the BCP and BCP-ACK may not be bound to a fixed processing delay, and the values given above merely represent an upper bound value.

V. Results

This section analyzes the performance of the different solutions proposed throughout the present paper and compares them with the simple shortest path routing with random wavelength assignment. To this end, simulations are conducted on the well-known NSFNET network composed of 14 nodes and 21 bidirectional links shown in Fig. 4. In such a scenario, there are 16 bidirectional wavelengths per link with 10 Gbps per channel. Moreover, link lengths are set to the real physical distances, as shown in Fig. 4.

Regarding the setup of the hardware devices, the control packet processing time and the non-blocking matrix switching time are set to 10 μs and 5 μs respectively. To compensate the processing delay incurred by the BCP at the OBS control unit in the switching process, an FDL is placed at each input port of the node, as described in [7]. These FDLs are not used to resolve burst contentions. In fact, we assume in all the examples that the network neither has wavelength converters nor FDLs for such purposes. In such a case, we assume that burst transmissions are subject to the wavelength continuity constraint and therefore, contended bursts count as blocked.

With respect to the traffic characteristics, bursts are created at each node following a Poisson process with rate $1/\lambda$ and a fixed size of 100,000 bytes. For simplicity, the burst destination is uniformly distributed to all the remaining nodes of the network, so that the probability of a burst to be sent to any other node in the network is the same.

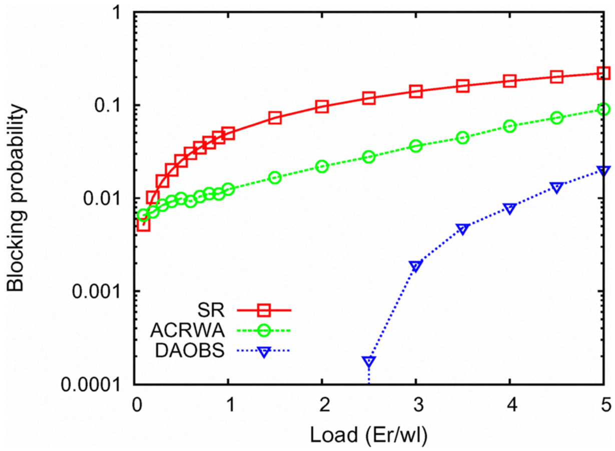


Fig. 5. Burst blocking on the NSFNET with 16 wavelengths.

Results are gathered using the batch means method after 1,600,000 bursts have been transmitted on the whole network. We have also obtained the confidence intervals, but as they are quite narrow, they have been omitted in order to improve the readability of the graphics.

The following notation is used in the graphs: the shortest plus random wavelength assignment scheme is denoted by SR, DAOBS is used for the distributed MAC protocol introduced in section III, and ACRWA for the protocol proposed in section IV. We evaluate the performance of the protocol in terms of, (1) burst blocking probability, (2) end-to-end delay and mean route lengths, and (3) link resources utilization.

In the specific case of ACRWA, the parameters were set to the following values: in the local transition rule, $\beta = 1.0$ and $r_0 = 0.9$; in the local updating rule we set $\alpha = 0$, i.e. the function does not modifies the tables values in the BCP forwarding; and finally, in the global updating rule, $\rho_1 = 0.001$. With respect to DAOBS, the size of the Local Queues (LQ) is fixed to 5 bursts for all the simulations.

A. Burst blocking probability

In this scenario we assess the burst blocking probability of the different schemes proposed in the paper. To this end, Fig. 5 gives a comparison of the three protocols, DAOBS, ACRWA, and SR as a function of the total offered network load expressed in erlangs per wavelength. Under normal circumstances and assuming that no contention resolution scheme is applied, SR offers the poorest results for all the load range. ACRWA improves the results with respect to SR, and DAOBS, the protocol that controls the resources assignment at the edge of the network, outperforms both.

ACRWA obtains results between half and one order better than SR for almost all the load range. The protocol not only improves the routing process in the core of the network for switching bursts, but also enhances the initial RWA in the

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792

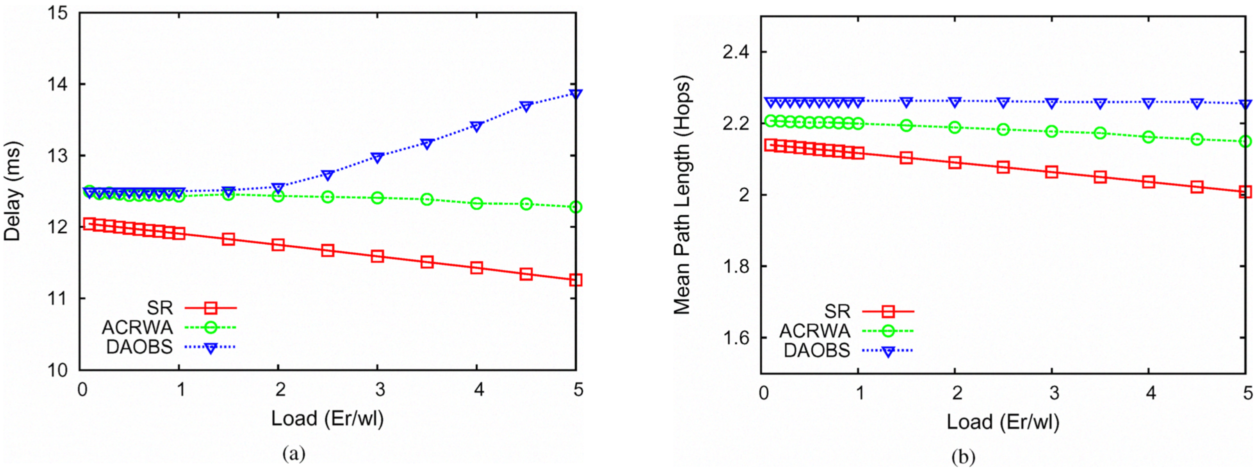


Fig. 6. (a) End-to-end delay, and (b) mean path length (in hops) on the NSFNET with 16 wavelengths.

transmission of them by assigning a wavelength and output port with a greater probability of success.

Regarding DAOBS, this protocol offers loss-free at low-to-medium load, while at medium-to-high load, there exist losses. Nonetheless, all these losses occur at the edge of the network due to the limited capacity of the ingress buffers, as mentioned above. The counterpart for such good behavior of DAOBS regarding burst losses is the extra delay introduced in the channel access as shown in the next subsection.

B. Delay and mean route length

Next results deal with the end-to-end delay experienced by bursts successfully received at the destination and its relationship to the mean traveled route length. Figure 6a shows the results of the delay (in ms) as a function of the total offered network load in erlangs per wavelength. Starting with the basic SR protocol, in this case we can see that the delay decreases when the network load increases. Due to the great number of burst losses in SR, specially at high loads, only bursts that travel along a shorter path and consume less resources arrive to the destination, as shown in Fig. 6b. Thus, as the mean path length is shorter, the propagation delay also decreases.

Regarding the ACRWA protocol, for all the network load, its delay is greater compared to SR. This is related also to the longer mean path length used by bursts in ACRWA as a result of distributing the burst traffic flows over a greater number of routes and wavelengths in the resources assignment process in the core of the network. Similar to the SR, in ACRWA the delay and mean path length also decreases, more slightly though.

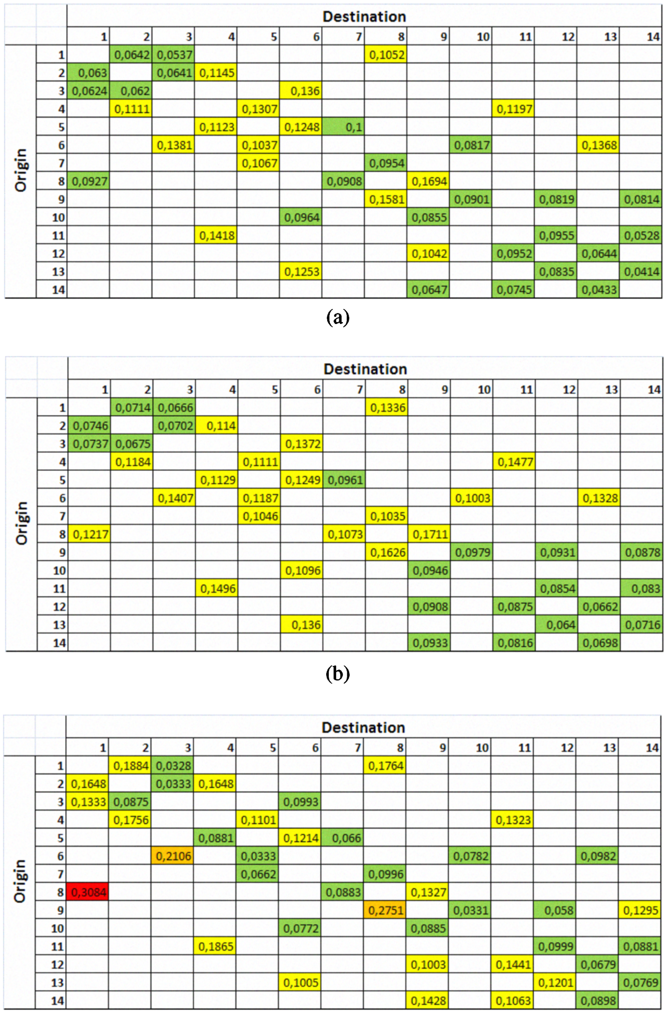
Finally, DAOBS gets the worst performance in terms of end-to-end in accordance to the greatest mean path length of the three analyzed protocols. The path length is nearly constant for all the load range which can be explained by the use of the same wavelength tree instantiation scheme throughout all the simulations. In our case, we used an own-made protocol based on shortest spanning trees and the connectivity degree offered by unused links to color the network graph and instantiate on it the DAOBS trees. Besides, we see that the delay rises when the load on the network also increases. This behavior is closely linked to the extra delay that bursts need in order to access the channel. When the DAOBS tree is highly loaded, the number of free slots for transmission tend to scarce, hence nodes need more time to get permission to transmit.

C. Link resources utilization

Last section of the results analyzes the link resources utilization for the three same protocols at an offered network load of 2 erlangs per wavelength. Figure 7 shows three tables which correspond to the matrix link connectivity of the NSFNET network. Numbers on the left column are for the origin node of the link, and those on the top row denote the destination node of it. A number of cells are cleared, that is, there is no link between the two network nodes. The rest of cells are colored according to the degree of link utilization. The graded utilization rate assuming a total offered network load of two erlangs goes from low ($\rho \leq 0.1$) (green), through low-medium ($0.1 < \rho \leq 0.2$) (yellow) and medium ($0.2 < \rho \leq 0.3$) (orange), to high ($\rho > 0.3$) (red). Utilization rates are also given in order to improve the readability of the figure in absence of color printing.

If we compare the results from SR (Fig. 7a) and ACRWA (Fig. 7b) we can see that both architectures lead to a similar resource utilization, although with some noticeable variations. A correlation coefficient of 0.911 indicates us that the strength and direction of the linear relationship between the SR and ACRWA link utilization data series is very strong. Nonetheless, we can see that in ACRWA we have a greater number of low-medium loaded links and that the utilization of low-loaded links is more distributed. This is not only a result from the lower number of losses, which implies a greater channel utilization, but also from the better resources assignment in the burst routing and switching which is done over a more diverse number of routes.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792



		Destination													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Origin	1		0,0642	0,0537					0,1052						
	2	0,063		0,0641	0,1145										
	3	0,0624	0,062				0,136								
	4		0,1111			0,1307						0,1197			
	5				0,1123		0,1248	0,1							
	6			0,1381		0,1037					0,0817			0,1368	
	7					0,1067			0,0954						
	8	0,0927						0,0908		0,1694					
	9								0,1581		0,0901		0,0819		0,0814
	10						0,0964			0,0855					
	11				0,1418								0,0955		0,0528
	12									0,1042		0,0952		0,0644	
	13						0,1253						0,0835		0,0414
	14									0,0647		0,0745		0,0433	

(a)

		Destination													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Origin	1		0,0714	0,0666					0,1336						
	2	0,0746		0,0702	0,114										
	3	0,0737	0,0675				0,1372								
	4		0,1184			0,1111						0,1477			
	5				0,1129		0,1249	0,0961							
	6			0,1407		0,1187					0,1003			0,1328	
	7					0,1046			0,1035						
	8	0,1217						0,1073		0,1711					
	9								0,1626		0,0979		0,0931		0,0878
	10						0,1096			0,0946					
	11				0,1496								0,0854		0,083
	12									0,0908		0,0875		0,0662	
	13						0,136						0,064		0,0716
	14									0,0933		0,0816		0,0698	

(b)

		Destination													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Origin	1		0,1884	0,0328					0,1764						
	2	0,1648		0,0333	0,1648										
	3	0,1333	0,0875				0,0993								
	4		0,1756			0,1101						0,1323			
	5				0,0881		0,1214	0,066							
	6			0,2106		0,0333					0,0782			0,0982	
	7					0,0662			0,0996						
	8	0,3084						0,0883		0,1327					
	9								0,2751		0,0331		0,058		0,1295
	10						0,0772			0,0885					
	11				0,1865								0,0999		0,0881
	12									0,1003		0,1441		0,0679	
	13						0,1005						0,1201		0,0769
	14									0,1428		0,1063		0,0898	

(c)

Fig. 7. Link resources utilization at an offered load of 2 Er: (a) in SR, (b) in ACRWA, and (c) in DAOBS.

Figure 7c illustrates the results for the DAOBS case. As shown, these are totally different to the previous ones. The distribution of paths to transmit the burst using DAOBS trees is radically different, and in previous cases where some links were rather used, now become overused in the transport of bursts. In fact, the correlation coefficient between DAOBS and ACRWA is 0.432, and even lower for the case between DAOBS and SR which is 0.356. For instance, the link between nodes 1 and 8 is now intensively used, acting as a connector between the nodes of the west and east coasts from the NSFNET network, and as this link is one of the longest ones on the network, its propagation delay rises the mean end-to-end delay as it is shown in Fig. 6a.

VI. Conclusion

In this paper we have presented two different approaches that deal with the resources assignment for optical burst-switched networks in a distributed way, one at the edge and the other one in the core of the network. On the one hand, DAOBS is a MAC protocol for OBS networks aiming at controlling the access transmission of the burst to avoid losses in the network core. It offers loss-free at low-to-medium traffic loads, and only at high loads, losses arise due to the saturation of the ingress buffers. As a counterpart, the mean access delay is slightly increased.

On the other hand, ACRWA is a routing and wavelength assignment protocol that enhances the routing assignment for switching bursts in transit, as well as it improves the wavelength assignment in the burst transmission process. The protocol uses BCPs for making the signalling and the reservation of resources along the route, and ACK-BCPs to update the state information tables which are used by the algorithm to make the resources assignment. Simulation results show that ACRWA offers a burst blocking probability improvement of nearly half order for almost the whole simulated load range with respect to the shortest path routing with random wavelength assignment protocol.

As future research lines, we would like to improve the wavelength or bus assignment process of the DAOBS protocol in order to minimize the mean access delay. And regarding ACRWA, we plan to make a study of the performance of the protocol under failure scenarios to test the resiliency and robustness capabilities of the protocol.

Acknowledgment

The authors would like to thank the workshop committee for their kind invitation to submit the present paper. The work in this article has been supported by the Spanish "Ministerio de Educación y Ciencia" and FEDER through project TSI2006-12507-C03-03, the European Commission within the Euro-NF NoE and by the Government of Catalonia and the European Social Funding through a pre-doctoral scholarship.

References

[1] Y. Chen, C. Qiao, and X. Yu, "Optical Burst Switching: A New Area in Optical Networking Research", *IEEE Network*, vol. 18, no. 3, pp. 16-23, May 2004.

[2] A. Agustí, C. Cervelló-Pastor, and M.A. Fiol, "A general resource assignment scheme for successful transmission in optical burst switched networks", *Optical Switching and Networking*, vol. 5, no. 4, pp. 232-243, 2008.

[3] A. Gumaste, et al. "Light-mesh - A pragmatic optical access network architecture for IP-centric service oriented communication", *Optical Switching and Networking*, vol. 5, no. 2-3, pp. 63-74, June 2008.

[4] J. Triay, C. Cervelló-Pastor, "Distributed Contention Avoidance in Optical Burst-Switched Ring Networks", *Proc. 2008 11th Int. Conf. on Communication Systems (ICCS)*, pp. 715-720, Guangzhou, China, Nov. 2008.

[5] J. Triay, and C. Cervelló-Pastor, "An Ant-based Algorithm for Distributed RWA in Optical Burst Switching", *Proc. 2009 11th Int. Conf. on Transparent Optical Networks (ICTON)*, Azores, Portugal, June 2009.

[6] M. Dorigo and T. Sttzle, *Ant Colony Optimization*, The MIT Press, Massachusetts, 2004.

[7] M. Klinkowski, D. Careglio, and J. Solé-Pareta, "Comparison of Conventional and Offset Time-Emulated Optical Burst Switching", *Proc. 2006 8th Int. Conf. on Transparent Optical Networks (ICTON)*, pp. 47-50, Nothingham, UK, Jun. 2006.

[8] G.S. Pavani, and H. Waldam, "Traffic Engineering and Restoration in Optical Packet Switching Networks by means of Ant Colony Optimization", *Proc. of 2006 3rd Int. Conf. on Broadband Communications, Networks, and Systems (BROADNETS)*, pp. 1068-1077, Oct. 2006.

[9] Z. Shi, Y. TinJin, and Z. Bing, "Ant algorithm in OBS RWA", *Proc. of SPIE, Optical Transmission, Switching and Subsystems II*, vol. 5625, pp. 705-713, Feb. 2005.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7792
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7792