Distributed Routing Path Optimization for OBS Networks based on Ant Colony Optimization

João Pedro, João Pires, and Joao Paulo Carvalho

1 Nokia Siemens Networks Portugal S.A., R. Irnãos Siemens 1, 2720-093 Amadora, Portugal
2 Instituto de Telecomunicações, Instituto Superior Técnico, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
3 INESC-id, Instituto Superior Técnico, R. Alves Redol 9, 1000-029 Lisboa, Portugal
joao.pedro@nsn.com

Abstract—This work proposes a distributed framework for routing path optimization in Optical Burst-Switched (OBS) networks loosely mimicking the foraging behavior of ants, which in the past has originated the Ant Colony Optimization (ACO) metaheuristic. The distributed framework consists of additional data structures stored at the nodes and special control packets used to estimate the goodness of the routing paths and update the routing tables of the nodes. The performance of the ACO-based framework is evaluated through network simulation, using two reference network topologies and compared with that obtained with shortest path routing and centralized routing path optimization. The simulation results show that the distributed framework significantly improves the performance of OBS networks, when compared to that of using shortest path routing, and attains a comparable performance to that of the centralized strategy. Moreover, the results also suggest that the framework is robust, as it does not require fine tuning its main parameters.

Keywords—ant colony optimization, contention minimization, optical burst switching, routing path optimization

I. INTRODUCTION

The long-haul and metropolitan networks deployed today exploit the high transmission capacity provided by optical fiber and WDM technology [1], but optical switching is performed at the wavelength channel granularity (e.g. 10 Gb/s, 40 Gb/s), which hampers their effectiveness to cope with bursty Internet Protocol (IP) packet traffic. Research efforts have been directed to finer-grained optical networking paradigms, such as optical burst switching [2]. At the ingress nodes of an OBS network, multiple IP packets directed to the same egress node are assembled into bursts of data, thus relaxing the control and switching overheads, when compared to transmitting a single IP packet. Moreover, because the resources for transmitting a burst are reserved using a Burst Header Packet (BHP), sent out-of-band and in advance of the data burst and processed electronically at each core node, OBS networks can avoid using complex and immature optical processing capabilities.

OBS networks rely on a one-way resource reservation mechanism to allocate wavelength channels to bursts [3]. By initiating data burst transmission without waiting for an acknowledgement of successful resource reservation along the routing path of the burst, data transfer delay can be reduced, when compared to that of using a two-way resource reservation mechanism. However, with one-way resource reservation there is the possibility that two or more bursts contend for the same wavelength channel at the core nodes. Unresolved contention results in data burst loss, degrading network performance.

In order to mitigate burst loss, OBS networks can employ contention resolution strategies, which overcome contention when it takes place [3], or contention minimization strategies, which reduce the probability of contention [4]. Wavelength conversion is the preferred contention resolution strategy for OBS networks and in most proposals it is commonly assumed that full-range wavelength converters are deployed at the OBS core nodes. Fiber Delay Line (FDL) buffers [5] and deflection routing [6] have also been proposed to further improve the contention resolution capability of the nodes.

Albeit some recent efforts have developed contention minimization strategies that exploit the wavelength domain [4] or jointly the wavelength and time domains [7], the majority of strategies for minimizing contention are based on balancing the burst traffic load across the fiber links [8]–[13]. This can be accomplished by optimizing the routing paths of the data bursts so as to decrease the traffic load on the most congested links at the expense of increasing it on less congested ones. Besides being effective in proactively reducing contention, routing path selection strategies for OBS networks should also be scalable, robust, and quickly adaptable to changes in the traffic pattern.

This paper describes a distributed framework for routing path optimization in OBS networks borrowing concepts from ant colony optimization. By means of network simulation using different network topologies, it is shown that the proposed framework efficiently reduces contention, reaching a blocking performance comparable to that of a centralized routing path selection algorithm. Moreover, it is shown that the framework is robust, as it does not demand fine tuning its main parameters.

The remainder of the paper is organized in the following way. Section II gives an overview of the existing routing path optimization strategies for OBS networks. Section III describes the proposed distributed routing path optimization framework for OBS networks, which embeds problem solving capabilities observed in colonies of some ant species. Section IV evaluates, through network simulation, the performance obtained with the proposed routing path selection framework and compares it with that obtained with shortest path routing and a centralized algorithm for optimizing the routing paths. Finally, concluding remarks are presented in section V.
II. ROUTING PATH OPTIMIZATION IN OBS NETWORKS

Strategies for optimizing the routing paths in OBS networks have received considerable attention in recent years. In these works, bursts are not necessarily routed through the shortest path, but instead go through paths that are expected to balance the traffic load across the network links, decreasing congestion and reducing the burst blocking probability. Load balancing becomes particularly important in OBS networks because, unlike with electronic packet networks, they cannot rely on efficient buffers at the core nodes for relieving congestion. The routing path optimization strategies for OBS networks can be broadly classified into centralized and distributed.

A. Centralized Routing Path Optimization

Centralized strategies rely on collecting and conveying to a single point of the OBS network long-term network-wide information and using it for optimizing offline the routing paths. This information basically consists of both the network topology and the average traffic load values that are expected to be offered between each pair of nodes. These average values are assumed to be obtained empirically or based on long-term predictions of the network load.

The routing strategies proposed in [8]–[10] use as inputs the network topology and average offered traffic load values to determine a single primary path for each node pair. The work in [8] uses a modified Dijkstra algorithm, whereas in [9] a tabu-search based algorithm is used to optimize the primary path selection. Both algorithms account for the expected traffic load distribution across the network links and aim to reduce the burst traffic load on the most congested links. Finally, [10] proposes a more elaborated, but rather complex, Integer Linear Programming (ILP) model to find the primary paths. For that purpose, the average burst losses at the fiber links are estimated using the Erlang-B formula. Still, given the non-linear nature of the problem [10], the proposed model includes a number of approximations to reduce the original problem to a linear form.

In all three strategies, the routing paths can be recomputed in response to significant changes in the network topology or the traffic pattern, as long as these modifications take place over relatively long time scales. Still, the major drawback of centralized routing path selection strategies comes from the need to collect all the information on a single point of the network, compute the routing paths and then disseminate them to the network nodes. Clearly, this process limits the ability to quickly adapt the routing paths to changes in the network and traffic conditions.

B. Distributed Routing Path Optimization

Distributed routing path optimization strategies are often preferred because, in contrast to the centralized approach, the congestion information collected by the network nodes is sent to the ingress edge nodes, which exploit this information to determine, in parallel with the other ingress nodes, the routing paths used to carry the locally assembled burst traffic to all the egress nodes. Hence, the distributed strategies are expected to react faster to changes on the network and traffic conditions than the centralized strategies. Nevertheless, since routing path selection is made by each ingress node without coordination with the other ingress nodes, these nodes are prone to make “greedy” decisions, that is, they will likely select the best paths they can for the bursts they assemble without considering the overall optimization of the network performance. This can result in a worst network performance than that obtained with the centralized strategies. Moreover, the distributed routing path selection strategies demand the frequent dissemination of network state information, which is carried through the control channels. Consequently, these strategies are expected to have a larger control overhead than the centralized strategies.

Although there are many proposals for distributed routing path selection in OBS networks, in the following only three works are considered, which are representative of the main features of the existing strategies. The work in [11] describes two distributed routing path selection algorithms. The first uses pre-computed candidate paths ordered according to a given criteria (e.g., increasing hop count) and assumes that the core nodes periodically broadcast the status (congested or not congested) of their output links. A link is said to be congested if the traffic load offered to it exceeds a given threshold. Based on the status of the links, the ingress nodes select the first path without congested links. In case all paths include a congested link, the algorithm breaks ties by selecting the one with smaller number of congested links. The second algorithm uses the periodically broadcasted offered link loads and eventually other metrics like the physical distance, to assign weights to the links and then determines, by means of the Dijkstra algorithm, a least-congested routing path between each pair of nodes.

The work in [12] proposes keeping a set of pre-computed candidate paths per node pair at the ingress nodes and transmitting the burst traffic through the routing path with the lowest burst blocking probability in the recent past. The burst blocking probability of each candidate path is updated by receiving, through the control channel, Acknowledgment (ACK) and Negative ACK (NACK) messages that signal the success or failure of each transmission, respectively. Although most of the traffic is transmitted via the least congested path, a small amount is sent through each of the pre-computed paths, in order to monitor their blocking probability. Alternatively to sending part of the bursts on the more congested paths, the work in [13] proposes transmitting all data bursts by the least congested path and, at the same time, sending search packets, carrying a fictitious reservation, on the remaining candidate paths. Although these packets are forwarded like regular BHPs, the control unit of the nodes processes a search packet and determines the resource availability for its fictitious data burst, but does not actually allocate resources for it. The blocking probability of each candidate path is monitored from ACKs and NACKs received from the transmission of both data bursts and search packets. However, this strategy increases the control overhead, when compared to that of [12].

III. DISTRIBUTED ANT-INSPIRED BURST ROUTING

Ant colony optimization is based on the foraging behavior of some ant species and has become one of the most successful swarm intelligence techniques [14]. Despite the individual simplicity of ants, when acting collectively they are able to execute complicated tasks with significant consistency and reliability. These relatively complex tasks seem to emerge from
interactions between large numbers of ants and their environment. In some cases, individual ants coordinate their activities using a principle known as stigmergy [15], an indirect and non-symbolic form of communication mediated by the environment. The communication between ants is based on the use of chemicals called pheromones. Particularly important for some ant species is the trail pheromone, which they use for marking paths on the ground. While walking between food sources and the nest, ants deposit pheromones on the ground, forming a pheromone trail. Subsequent ants sense pheromone and tend to choose, probabilistically, the paths with stronger pheromone concentration, further reinforcing it. After some time, ants converge to use a single shortest path [16].

Given the intrinsically distributed nature of ACO, it has found application in the routing problems of communications networks. All ACO algorithms for network routing have in common the fact that pheromone tables are stored at the nodes and artificial agents, typically in the form of control packets, are used to collect node/link state information and update the pheromone tables. Two of the most relevant ACO applications for network routing are ABC [17], which was developed for routing in circuit-switched networks, like the Plain Old Telephone Service (POTS), and AntNet [18], designed for packet-switched networks, such as IP networks.

Despite the recent research activity aiming to also use ACO for routing in optical WDM networks [19], [20], to the best of the author’s knowledge ACO has not yet been adapted to the particular case of OBS networks. The distributed routing path optimization framework for OBS networks described in this work, which is designated as Distributed Ant-inspired Burst Routing (DABR), inherits some concepts from both ABC and AntNet, but was specifically designed for bufferless OBS networks and incorporates functions that account for the particular characteristics of these networks.

Common to all applications of ACO for network routing, the DABR framework uses artificial ants, in the form of control packets, and data structures at the network nodes to support stigmergic communication between the artificial ants. The key features of DABR are:

- Forward ants carry a fictitious data burst reservation and, at each node, the instantaneous congestion of the output fiber links is evaluated based on the number of wavelengths available to transmit the fictitious burst;
- Backward ants update the pheromone tables using the congestion information collected by the forward ants as input to a loss performance model that estimates the burst blocking probability on the path traversed by the correspondent forward ant;
- In order to mitigate path stagnation, three mechanisms are used: the routing of forward ants employs both a minimum pheromone concentration per adjacent node (noise) and pheromone-heuristic control, whereas the backward ants use privileged pheromone laying [21];
- The nodes keep separate routing and pheromone tables because the routing tables must be updated using a path integrity mechanism capable of completely avoiding the formation of circular routing paths (loops).

Consider an OBS network modeled as a directed graph $G = (V, E)$, where $N = |V|$ is the number of nodes and $L = |E|$ is the number of unidirectional fiber links. Assume each fiber link supports a set of $W$ wavelengths, $\{\lambda_1, \lambda_2, \ldots, \lambda_W\}$. The following sub-sections detail the DABR framework.

A. Control Packets and Local Data Structures

In DABR, artificial ants are transmitted on the control channel as modified BHPS. Two types of artificial ants are used: the Explorer Ant Packet (EAP) and the Referee Ant Packet (RAP). The former ant is used to collect information on the resource availability along a given routing path and to update the pheromone tables, loosely mimicking the behavior of real ants, whereas the latter one is used to avoid circular path formation. Both EAP and RAP have two modes of operation: forward (F), from the ingress node to the egress node, and backward (B), in the reverse path. Two data structures are stored at each node: one pheromone table and one routing table. The pheromone table is used for routing the F-EAPs and it is updated in their return trips as B-EAPs. The routing table can only be updated by RAPs.

Unlike previous ACO-based routing algorithms, routing of artificial ants and data bursts in DABR is based not only on the egress node but also on the ingress node, giving an additional degree of freedom for load balancing at the expense of increasing the size of the tables and eventually demanding more artificial ants. Although this approach increases the complexity of deploying the DABR framework in an OBS network, it can provide significant performance improvements, as will be confirmed in the performance evaluation.

Let $A_i$ denote the set of adjacent nodes of $i \in V$ (that is, nodes connected to $i$ by a fiber link) and let $\tau_{ij}^{sd}$ denote the pheromone value associated to using $j \in A_i$ to route at node $i$ a burst generated at $s \in V$ and directed to $d \in V$. The sum of the pheromone values related with the same ingress and egress node pair must equal one, that is,

$$\sum_{j \in A_i} \tau_{ij}^{sd} = 1, \quad i, s, d \in V, \ s \neq d . \quad (1)$$

Let $h_{ij}$ denote the number of links of the shortest path between $i$ and $j$. The pheromone tables are initialized as to be slightly biased towards the shortest paths,

$$\tau_{ij}^{sd} = \frac{1/(1 + h_{jd})}{\sum_{k \in A_i} 1/(1 + h_{kd})}, \ j \in A_i, \ i \neq d, \ s \neq d \quad 0, \text{ otherwise}. \quad (2)$$

B. Routing of Forward Explorer Ants

Each time node $s$ assembles a data burst directed to $d$, it generates with probability $p_{max}$ where $0 < p_{max} \leq 1$, an F-EAP that is routed from $s$ to $d$ to collect information on the resource availability of a particular path. The F-EAP carries a fictitious burst reservation request whose duration equals that of the burst that triggered the generation of the artificial ant and a list $V_{EAP}$ with the nodes it has already visited.

978-1-4244-4148-8/09/$25.00 ©2009
At each node $i$ visited by an F-EAP, the next node of its routing path is selected among the adjacent nodes of $i$ not visited yet. In case $A_i \subseteq V_E$, the F-EAP is dropped to avoid looping. Otherwise, the adjacent node $j$ is selected with a probability $p_{ij}^d$ that is the normalized sum of the pheromone value and a heuristic value $\eta_{ij}$ that accounts for the resource availability on the fiber link between nodes $i$ and $j$ [18],

$$p_{ij}^d = \frac{\tau_{ij}^d + \eta_{ij}}{\sum_{k \in A_i} \tau_{ik}^d + \sum_{k \in A_i \setminus \{i\}} \eta_{ik}}, \ j \in A_i, j \not\in V_E,$$

where \(\{v_1, v_2, ..., v_k, v_{k+1}\}\) denotes the ordered set of the nodes visited by the F-EAP and $\hat{B}_{i[v_{k+1}]}$ the estimated burst blocking probability in the fiber link between $v_k$ and $v_{k+1}$. The goodness of a routing path is quantified in DABR as,

$$q_{sd}^d = 1/B_{sd}^d.$$

Each node $i$ maintains the goodness of the best solution $q_{sd}^{best}$ estimated by the previous $Q$ artificial ants between $s$ and $d$ and the pheromone reinforcement value $r$ is then given by,

$$r = \begin{cases} 
q_{sd}^d / q_{sd}^{best}, & q_{sd}^d < q_{sd}^{best} \\
1, & \text{otherwise} \end{cases}$$

In order to enhance the rewarding of good solutions (high values of $r$), while saturating the rewards for low values of $r$, the following squash function $s(x)$ is used [18],

$$s(x) = \left(1 + \exp\left(\frac{1}{r |V_E|}\right)^{-1}\right)^{-1},$$

where $\tau_{ij}$ is the maximum pheromone value that a single artificial ant can deposit. This parameter is used to regulate the impact of a single artificial ant on the pheromone tables.

The pheromone update has to guarantee a minimum amount of pheromone $\tau_{min}$ uniformly distributed over all the adjacent nodes,

$$\tau_{ij}^d = \tau_{min} / \|A_i\|, \ k \in A_i,$$

while ensuring that equation (1) holds. Hence, the following expressions have been devised for updating the pheromone value associated to the adjacent node $j$ the F-EAP has used and that associated to the remaining adjacent nodes,

$$\tau_{ij}^d \leftarrow \tau_{ij}^d + r \cdot \left(1 - \frac{\tau_{ij}^d - \tau_{min}}{(|A_i| - 1) / |A_i|}\right),$$

$$\tau_{ik}^d \leftarrow \tau_{ik}^d - r \cdot \left(\tau_{ik}^d - \tau_{min}\right), \ k \in A_i, k \neq j,$$

In case the update has modified the adjacent node that has the highest pheromone value, designated as the dominant adjacent node, a specific flag of the B-EAP is activated.

### D. Routing Table Update and Circular Path Avoidance

In the initial implementation of DABR, data bursts were routed through the path obtained from the adjacent nodes with the largest pheromone concentration. However, it was found that circular paths were occasionally formed, which resulted in data bursts getting trapped on a loop. Although these paths are only temporary, the resulting communication disruption is not tolerable in a high capacity optical network. Therefore, it is imperative to include in DABR a mechanism to avoid the
the formation of circular paths. The novel mechanism devised for this purpose separates the pheromone and routing tables, guaranteeing that the update of the routing paths never allows the formation of loops without introducing any additional restrictions to the update of the pheromone tables. However, the mechanism requires additional control packets.

Let $v_{sd}^{j}$ denote the adjacent node of node $i$ used for routing data bursts from node $s$ to node $d$. The update of the routing path between $s$ and $d$ is triggered by the arrival at node $s$ of a B-EAP with the flag that signals a new dominant adjacent node activated. As a result, node $s$ generates one F-RAP directed to node $d$. This control packet includes a list to record the nodes it visits. At each node $i$ the F-RAP visits, the dominant adjacent node $j^*$ is determined as,

$$ j^* = \arg \max_{j \in A_i} v_{ij}^{sd}, $$

and compared with the nodes previously visited. In case node $j^*$ has already been visited, a loop is detected and the F-RAP is discarded, keeping the routing path unchanged. Otherwise, the F-RAP adds the dominant adjacent node $j^*$ to the list of nodes visited and moves to it. In case $j^* = d$, the new routing path is non-circular. Thus, the F-RAP is switched to B-RAP, follows the inverse path and, at each node $i$, the adjacent node used to route data bursts, $v_{id}^{j^*}$, is updated to be the same as the node visited by the F-RAP immediately after $i$.

The routing of data bursts in DABR is determined by the routing tables of the nodes. When a BHIP from $s$ to $d$ arrives at node $i$, this node selects the next node of the burst from $v_{id}^{j^*}$.

The distributed routing framework more similar to DABR is [13], which also uses fictitious data bursts to assess the goodness of the paths. As with most distributed algorithms for optimizing the routing paths in OBS networks, the work in [13] relies on burst blocking probability statistics to estimate the goodness of the routing paths. In order to differentiate the quality of two paths between the same pair of edge nodes, a number of blocking events must be observed. However, when both paths are lightly loaded, a large number of bursts have to be transmitted until a single blocking event occurs, which requires a long time to accurately differentiate the goodness of two or more lightly loaded routing paths. On the other hand, DABR can quickly differentiate the quality of multiple lightly loaded paths because it estimates the blocking probability of the paths based on the occupation of the fiber links they traverse instead of collecting blocking probability statistics.

IV. RESULTS AND DISCUSSION

The loss performance of OBS networks is evaluated using the network simulator of [23] modified to support DABR. All the simulations use the JET resource reservation [2], each fibre link has $W = 32$ wavelengths with a capacity $\mu = 10 \text{ Gb/s}$ each, the time to process one BHP or artificial ant is $100 \mu s$ and the switch configuration time is $160 \mu s$. Unless stated otherwise, the burst size and interarrival time are exponentially distributed and the average burst size is $10 \text{ MB}$. The average offered traffic load normalized to the network capacity is given by,

$$ r^{sd} = \frac{\sum_{x \in S} \gamma^{s,x} \cdot h^{s,x}}{L \cdot W \cdot \mu}, $$

where $r^{sd}$ is the average offered traffic load between nodes $s$ and $d$. The performance evaluation is made with the network topologies of Figure 1, assuming a uniform traffic pattern for EON and a non-uniform traffic pattern for COST 239. The average burst blocking probability and the correspondent 95% confidence interval are obtained from running 10 independent simulations with $5 \times 10^6$ data bursts each.

Figure 2 plots the average burst blocking probability as a function of the average offered traffic load for the COST 239 and EON networks, respectively. For comparison purposes the plots also include the loss performance obtained with Shortest Path Routing (SPR), the centralized HMNC algorithm [9], and a DABR implementation that does not differentiate routing of bursts by their ingress node, called Merged Routing Paths (MRP) in opposition to the original Individual Routing Paths (IRP). The parameters used for both DABR implementations are $p_{\text{at}} = 0.05$, $\alpha = 0.25$, $\tau_{\text{min}} = 0.20$, $\tau_{\text{max}} = 0.10$, and $Q = 50$.

The simulation results show that the performance obtained with DABR lies between that of SPR and HMNC, although closer to the latter. This is an expected outcome. On one hand, the simpler SPR does not optimize the traffic load distribution over the links, which can significantly increase contention in some of them. On the other hand, the HMNC algorithm is more effective because it exploits the global knowledge of accurate traffic statistics. However, in highly dynamic traffic scenarios, where these statistics are less reliable, HMNC may become less effective than DABR. The IRP implementation of DABR outperforms MRP, showing that routing bursts based on both their ingress and egress nodes improves load balancing. In fact, for the EON network, DABR with IRP almost matches the performance obtained with HMNC. In the following, only the IRP implementation is considered. As a remark, note that the larger confidence intervals for DABR are due to the online modification of the routing paths, while with SPR and HMNC the same set of routing paths is used in the entire simulation.

Table I presents the average traffic load that can be offered to the OBS networks to match two values of the average burst blocking probability $B$. These load values are estimated by performing simulations with multiple values of $\Gamma$, determining the load values between which the value with blocking probability $B$ is located and using linear interpolation (with logarithmic scale for the average blocking probability). As shown, employing DABR and HMNC enables the network to sustain much more traffic load than that with traditional SPR.
Average offered traffic load, $\Gamma$

Average burst blocking probability $a$

SPR
DABR - MRP
DABR - IRP
HMNC

10^{-1}
10^{-3}
10^{-4}
10^{-2}
10^{-1}

(a) COST 239 network
(b) EON network

Figure 2. Network performance obtained with DABR, SPR, and HMNC.

TABLE I. AVERAGE OFFERED TRAFFIC LOAD FOR $B = \{10^{-3}, 10^{-4}\}$

<table>
<thead>
<tr>
<th>$B$</th>
<th>COST 239 network</th>
<th>EON network</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SPR</td>
<td>HMNC</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>0.115</td>
<td>0.353</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>0.094</td>
<td>0.295</td>
</tr>
</tbody>
</table>

The Erlang-B formula (5) used in DABR to estimate the average burst blocking probability is accurate for exponentially distributed burst interarrival time. In order to show that DABR is efficient with other distributions, Fig. 3 plots the blocking performance using the COST 239 network and a lognormal distribution with two values for the Coefficient of Variation (CoV), defined as the ratio between the standard deviation and the square root of the average burst interarrival time. As can be seen, the blocking probability increases with the traffic burstiness. Nevertheless, this occurs with DABR, but also with both SPR and HMNC. In fact, the performance degradation observed when traffic burstiness has a tenfold increase is of similar magnitude for DABR and HMNC, suggesting that the DABR framework can be efficiently used in OBS networks regardless of the burst interarrival time distribution.

Figure 3. Performance impact of lognormal-distributed interarrival time.

The impact on performance of $\tau_{\min}$, $\tau_{\max}$, and $Q$ is plotted in Figures 4, 5 and 6, respectively. The parameter values are the same as above with the exception of the one being varied.

Figure 4. Performance impact of $\tau_{\min}$ using the COST 239 network.

Figure 5. Performance impact of $\tau_{\max}$ using the COST 239 network.

978-1-4244-4148-8/09/$25.00 ©2009

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.
The blocking performance slightly deteriorates with the minimum pheromone concentration on the adjacent nodes. This is because the random component of routing EAPs becomes larger, increasing the chances of modifications in the routing paths. In the limit case $\tau_{\text{min}} = 1$, the network cannot optimize the set of routing paths. The average burst blocking probability is also slightly increased as EAPs are allowed to deposit more pheromone in a single trip. This behaviour is attributed to more frequent modifications of the routing paths as a consequence of increased fluctuations on the pheromone values. A smaller pheromone increment makes the routing paths less sensitive to the goodness of the solution found by a single ant, which means they are changed only when a significant number of EAPs have followed a given routing path and confirmed its superior quality. The use of a large $Q$ can improve performance as it enhances the accuracy of the pheromone update process.

Overall, given the relatively small range of variation of the average burst blocking probability, the results suggest that DABR is quite robust to changes in its main parameters. The results also suggest that a larger $Q$ and a smaller $\tau_{\text{min}}$ and $\tau_{\text{max}}$ slightly improve the loss performance. However, it should be stressed that the simulations consider a single traffic pattern (in terms of average offered traffic load), whereas in more dynamic traffic scenarios a very large $Q$ and very small $\tau_{\text{min}}$ and $\tau_{\text{max}}$ should be avoided because they can slow down the convergence to a new set of optimized routing paths whenever significant changes in the traffic pattern take place.

V. CONCLUSIONS

This paper has proposed a distributed framework based on ACO concepts for optimizing the routing paths in OBS networks. Simulation results have shown that the proposed framework can significantly improve network performance without needing to fine tune its parameters and without being constrained to a specific burst interarrival time distribution.

REFERENCES


