

# Analysis of Latency, Blocking Probability and Network Utilization for a Specific Routing and Spectrum Assignment Algorithm, in Elastic Optical Networks

**Teodora Kocevska, Pero Latkoski, Marko Porjazoski, Borislav Popovski**

Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University, 1000 Skopje, Macedonia  
{teodora, pero, markop, borop}@feit.ukim.edu.mk

---

*Abstract: The emergence of new ICT applications has triggered significant growth in data traffic volumes. As a result, novel flexible and adaptive concepts for optical network management has emerged, in order to meet the heterogeneous and variable traffic demands. Efficient spectrum usage has become one of the most important issues, since the optical spectrum is not an infinite resource. Today's WDM (Wavelength Division Multiplexing) technology is inappropriate, due to the fixed spectrum grid it uses. The elastic optical networks (EON) will overcome these problems. Finding an efficient RSA (Routing and Spectrum Allocation) algorithm is very important for resourceful use of the network. This paper extends the current research knowledge regarding the intelligent RSA algorithm with support for different Class of Service (CoS). It assumes a coexistence of two traffic categories in the network, along with a central Software Defined Networking (SDN) controller. Furthermore, a dynamic RSA algorithm is used with two controlling coefficients to determine the priority of different classes. The performance of the algorithm and the impact of the controlling coefficients on several performance metrics (e.g. average latency for data traffic, blocking probability, network utilization and network efficiency) has been examined through MATLAB simulations. The obtained results indicate some complex influence of the data traffic load and spectrum-controlling RSA coefficient over the performance metrics. The results can help network operators not only define different CoS in their network, but to also finely tune its utilization, blocking probability and data traffic latency, which is crucial for the emerging 5G communication networks.*

*Keywords: Elastic Optical Networks; Software Defined Networking; RSA; 5G; MATLAB*

---

# 1 Introduction

Today, communication networks are an inseparable part of people's lives. They provide support for various personal daily activities (work, social life, entertainment, etc.), as well as, for most of the industrial sectors (engineering, marketing, trade, medicine, business, etc.). The future will introduce even a more noticeable role of communication networks in everyday life. With 5G, every person and every device, from anywhere, at any time, is expected to have access to information and to be able to share data. Network operators in the future are expected to offer a number of services with different features, such as extended Mobile Broadband (eMBB), Ultra-Reliable Low-Latency Communications (URLLC) and massive Machine-Type of Communications (mMTC). This requires for a new network architecture and a new traffic engineering strategy in all parts of the communication network.

The present optical transport networks are based on DWDM (Dense WDM) technology. Although this is a relatively mature optical transmission technology, it is not suitable for traffic demands in future optical networks, because it uses a uniform division of the spectrum, i.e. fixed frequency grid. This approach leads to spectral inefficiency. A new adaptive and flexible concept for optical networks management is needed in order to deal with future trends: flexible resource allocation, heterogeneous traffic demands, energy efficient solutions and reliability.

Flexible optical networks, also known as elastic optical networks, use the spectrum by dividing it into a large number of narrow frequency slices. In this way, the elastic optical networks can use wider spectral channels in order to serve the traffic requirements that need high bit rates (400 Gb/s, 1 Tb/s and more).

This paper extends the research findings made in [1] by introducing a variety of network relevant performance metrics and by discussing the importance of spectrum management for all classes of traffic. The paper also assumes a SDN controller to optimize the usage of network resources, thus, the spectrum allocation will be based on the network state information and the characteristics of the previously served traffic requests. This central entity can orchestrate the requests that must be served immediately, with regard to the requests that can tolerate a certain delay. Such an orchestration is particularly important to control the blocking probability for the requests that must be served immediately, as well as, to provide efficient network utilization. In order to analyze, evaluate and validate the proposed algorithm, many simulations were conducted in a custom-made MATLAB software tool. The simulations evaluate the influence of the priority control mechanism on the average latency of the traffic, the blocking probability, network utilization and network efficiency.

## 2 Related Work

Since the introduction of the elastic optical networks concept, there have been numerous research on the subject. The literature on elastic optical networks shows a variety of approaches. Only a part of the recent work has been selected here.

Most of the research has focused on elastic optical inter-datacenter networks because they are targeting high-bandwidth and low-latency requirements. Since resource availability is considered as one of the main constraints in today's optical networks, failure recovery is extremely important. So far, several studies have been performed within these two topics [2]-[6]. The research on protection in Elastic Optical Networks (EONs) has become very popular, as well. The authors of [7] propose a new dedicated path protection mechanism for link-failure survivability in EONs. Further, they used modified version of the Dijkstra's algorithm for calculation of the link-disjoint routes. Other well-known algorithms were used for the sake of comparison with the proposed algorithm.

The RSA problem must be considered in order to properly analyze, design, plan, and operate flex-grid networks [8]. In recent years, the RSA issue has attracted considerable attention from research teams, since the development of efficient methods for solving realistic problem instances in reasonable time has become an imperative. Several papers investigated RSA in spectrum sliced EONs. The reference [9] reviews and classifies different routing and spectrum allocation approaches. Additionally, the authors presented fragmentation, modulation, quality-of-transmission, traffic grooming, survivability, energy saving, and networking cost, as important aspects related to RSA. Authors of [10] studied the problem complexity when different technological options for the EON realization are used. In [11], authors proposed a technique which improves resource allocation efficiency and lowers the call blocking for scheduled demands. Paper [12], focuses on minimizing the maximum spectrum usage for a set of traffic demands and proposes new cost functions that are pluggable into an auxiliary layered-graph framework to solve the routing, fiber, waveband, and spectrum assignment problem. The work in [13], proposes a dynamic RSA algorithm that uses Back Propagation Neural Network (BPNN) to predict the arrival time and the holding time of future arrival requests. Authors of [14] have designed an algorithm that minimizes the process of selection with priority-based encoding for selecting optimized path in the existing routes. In [15], a novel heuristic algorithm for the problem of routing, spectrum, transceiver and regeneration allocation is proposed. Moreover, a trade-off between spectrum usage and transceiver usage was investigated through simulations. In [16] two virtualized functions for routing and frequency allocation (RFA) are proposed and different ordering policies were investigated. The development of an efficient mechanism that is able to handle traffic requests that need to be served immediately and others that can be reserved in advance has gained importance in recent years. The coexistence of immediate and advanced reservation request is an important issue since advanced reservation

(AR) requests tend to reserve future resources which can cause a blocking of the immediate reservation (IR) requests. The concept of advanced reservation was first introduced in [17]. In [18], the authors summarized related research into EONs and resource-allocation methods for IR and AR requests, and proposed dynamic RSA method to reduce spectrum fragmentation, while controlling the service level of IR and AR requests. In [1], *Shahsavari et al.* proposed multi-quality of service routing and spectrum assignment algorithm in EONs considering the bandwidth usage and the holding time of established connections. Moreover, authors suggested use of two controlling coefficients to determine the ratio of time and spectrum slots which can be assigned to big data traffic (BDT) requests and demonstrated the influence of this coefficients on the blocking probability.

In this paper, we take into consideration, the previously mentioned related work, especially the work provided in [1], and extend the research in the area of network utilization and data latency, as two key metrics related to the network management, in addition to the blocking probability.

### 3 Problem Description

Traffic demands that arrive to a network, can be classified into two categories: Priority Real-time Traffic (category 1) and Big Data Traffic (category 2). Finding an algorithm that will reduce the blocking probability of the first category traffic is essential since 5G communication networks are expected to support critical services such as self-driving vehicles, factory automation, remote health care (generally 5G URRLC and eMTC). Moreover, it is necessary to find a strategy for parallel handling of data traffic which can tolerate some delay (e.g. 5G eMBB, exchange of data between data centers, etc.). On the other hand, it is very important to improve the efficiency in the network resources utilization. In the case of optical networks, it is particularly difficult to support the coexistence of these two categories of traffic, since data traffic while more robust, can increase the blocking probability of the category 1 traffic.

The solution lays in the usage of a logically centralized SDN controller that has a global picture of the network state, based on which it can make strategic changes in the routing and spectrum allocation. This centralized entity can orchestrate the traffic demands from different CoS and improve the overall service. The focus of this paper is finding a way to determine the priority of the traffic requirements belonging to the first category, in relation to those of the second category, based on network conditions. An efficient method and performance metrics have been proposed in order to allow the network controller to handle traffic requirements that have different quality of service requirements.

### 3.1 Network Model

The network model discussed in this paper is an extended and improved version of the network model presented in [1]. Graph  $G(J, L)$  is used to denote the elastic optical network, where  $J$  is the set of switching nodes and  $L$  is the set of optical links between the nodes in the set  $J$ . The number of nodes in the network is denoted as  $N_j = |J|$ , and the number of bidirectional links is denoted as  $N_l = |L|$ . Each link is bidirectional, composed of a pair of optical fibers, one for each direction. Each optical fiber has capacity  $C$ . The optical fiber capacity is divided into  $N_{fs}$  frequency slices, where each frequency slice has capacity equal to  $M \times C / N_{fs}$ , where  $M$  represents the modulation level (in this paper BPSK is taken into account,  $M=1$ ). Each optical fiber can be represented as ordered set  $F = \{f_1, f_2, f_3, \dots, f_{N_{fs}}\}$  of frequency slices. The value of  $N_{fs}$  is set by the network designer.

The time duration of each connection is expressed in the number of time slots, because the usage of the spectrum is divided into time slots ( $TS$ ). Each time slot lasts  $T$  seconds.

Two types of requests are arriving in the network: requests that do not tolerate delay and must be served immediately (Fixed Rate Immediate traffic), denoted as FRI and requests that can tolerate delay (Bulk Data Traffic), denoted as BDT. Each FRI request is represented as  $f(s, d, t_a, n_{rfs})$ , where  $s$  is the source node index,  $d$  is the index of the destination node,  $t_a$  is the time point in which the request arrives in the network, and  $n_{rfs}$  is the number of required frequency slices. The service duration is not known in advance and the allocated bandwidth is released when the service is terminated. The BDT requests are denoted as  $b(s, d, t_s, t_e, n_{rb})$ , where  $s$  is the source node index,  $d$  is the index of the destination node,  $t_s$  is the arrival time of the request,  $t_e$  is the latest time moment in which the request can be accepted in the network and  $n_{rb}$  is the number of required resource blocks. The maximum tolerable delay after which the request is blocked is denoted as  $t_e - t_s$ . The duration of the BDT requests is specified in advance.

### 3.2 Simulation Scenario

A custom-made, MATLAB simulation tool is used for the analysis, evaluation and validation of the proposed algorithm. In particular, each network node is connected with the SDN controller that controls and optimizes the spectrum allocation. It is assumed that the SDN controller has all the information about the network state (bandwidth usage and duration of the established connections). For the purposes of the simulation, the topology of the National Science Foundation Network (NSFNET), with 14 nodes and 22 bidirectional links [19], has been used. The NSFNET topology is shown in Figure 1.

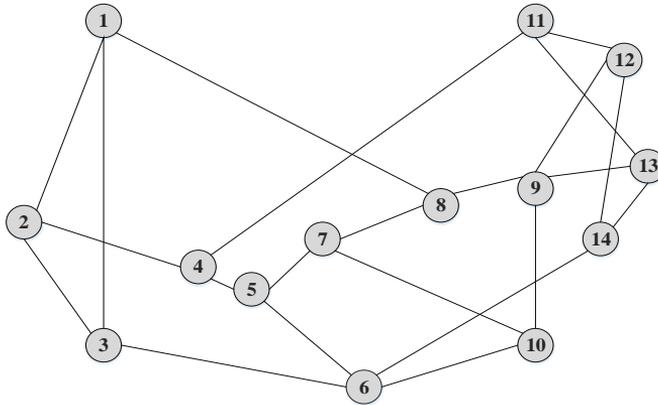


Figure 1  
NSFNET topology

Each optical fiber has a capacity  $C=2.5$  THz corresponding to  $N_{fs}=200$  frequency slices with a bandwidth of 12.5 GHz (for  $M=1$ ). In time domain, it is assumed that the duration of one TS is  $T=0.01$ s [1].

All traffic demands are generated from uniformly distributed sources and are addressed to uniformly distributed destinations. The arrival time of traffic demands is generated using the Poisson model with a mean duration of  $100 T$ . The number of frequency slices required by the FRI traffic demands is uniformly distributed in the range from 1 to 17. The FRI traffic load is fixed at 60 E (Erlang). The number of resource blocks in the time-frequency domain and the maximum tolerable delay in number of time slots are uniformly distributed within [100, 600] and [400, 700] intervals, respectively. The simulations are repeatedly executed for five values of the BDT traffic load (50, 75, 100, 125 and 150 E).

The average duration of FRI connections, as well as the average number of frequency slots assigned to the FRI connections, are calculated based on 100 served FRI demands. In all simulations, the following performance metrics are analyzed: average latency for BDT traffic, blocking probability for both traffic types, network utilization and network efficiency.

## 4 Routing and Spectrum Allocation Algorithm with Support for QoS

### 4.1 Symbols Used

The symbols used in the algorithm described below are summarized in Table 1.

Table 1  
Used symbols

Symbol	Description
$s$	Source node index
$d$	Destination node index
$K$	Number of shortest paths between $s$ and $d$
$C$	Optical fiber capacity
$M$	Modulation level
$N_j$	Number of nodes in the network
$N_l$	Number of bidirectional links in the network
$N_{fs}$	Number of frequency slices
$T$	Duration of one time slot (in seconds)
$n_t$	Number of simulated time slots
$t_a$	Arrival moment of the FRI request
$n_{rfs}$	Number of required frequency slices by the FRI request
$n_{ts}$	Number of required time slots by the FRI request
$t_s$	Arrival moment of the BDT request
$t_e$	Latest time moment in which the BDT request can be accepted in the network
$n_{rb}$	Number of required resource blocks by the BDT request
$q_{time}$	Control coefficient $0 \leq q_{time} \leq 1$
$T_{F,i}$	Duration of the $i^{\text{th}}$ FRI connection
$N_F$	Number of FRI connections that are taken into account in the calculation of the average duration of the served FRI connections and the average number of frequency slots assigned to the served FRI connections
$R_{avt}$	Average duration of the FRI connections
$R_t$	Maximum number of time slots that can be assigned to a BDT request
$q_{freq}$	Control coefficient $0 \leq q_{freq} \leq 1$
$L_{F,i}$	Number of frequency slices assigned to the $i^{\text{th}}$ FRI connection
$R_{avf}$	Average number of frequency slices assigned to the FRI connections
$R_f$	Maximum number of frequency slices that can be assigned to a BDT request

$t_{backoff}$	Backoff interval
$numBlockFri$	Number of blocked FRI requests
$numBlockBdt$	Number of blocked BDT requests

## 4.2 RSA Algorithm Description

Requests arriving in the network are represented by the set  $D$  defined with the Equation (3), which is a union of the  $FRI$  requests  $F$  (Equation (1)) set and the set of  $BDT$  requests  $B$  (Equation (2)). Each request is actually a connection between a pair of nodes in the network.

$$F = \bigcup_{\substack{s,d \in J \\ t_s \in \{1,2, \dots, n\} \\ n_{rfs} \in Z^+}} f(s, d, t_a, n_{rfs}) \quad (1)$$

$$B = \bigcup_{\substack{s,d \in J \\ t_s \in \{1,2, \dots, n\} \\ n_{rb} \in Z^+}} b(s, d, t_s, t_e, n_{rb}) \quad (2)$$

$$D = F \cup B \quad (3)$$

During the initialization of the simulation controlling process, all the parameters obtain values and the process goes to the state WAIT expecting to arrive either a FRI or a BDT request. If a FRI request arrives, the process goes to the SERVE\_FRI state, and if a BDT request arrives, the process goes to the SERVE\_BDT state, as is shown in Figure 2.

All requests that arrive in the network are placed in a queue, which is represented by a matrix. Matrix rows are sorted according to time slots (e.g.  $TS_1, TS_2, \dots, TS_{n_t}$ ) in which the requests are arriving. Same type request that arrive in a time slot are recorded one after another in the matrix. If multiple requests, that belong to different categories arrive within the same timeslot, the FRI requests are served before the BDT requests.

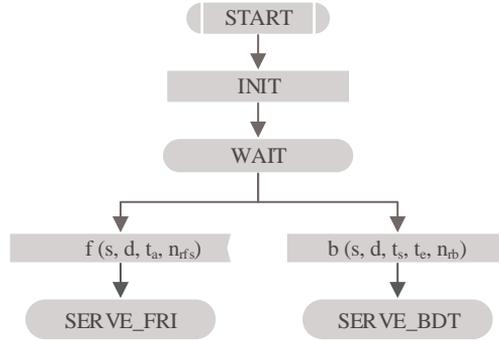


Figure 2

Formal description of the controlling process (initial transition)

The considered network has 22 bidirectional links and each link is composed of a pair of optical fibers. Further, each optical fiber (44 optical fibers) is represented by a  $N_{fs} \times n_t$  matrix. The columns represent the frequency slices of the optical fiber spectrum, and the rows are representing the time slots in which the frequency slices are used. The set of links can be represented as a 3D matrix of dimensions  $N_{fs} \times n_t \times 2N_l$ .

For each pair  $(s, d)$ , there are  $K$  shortest paths that can be calculated using an algorithm for finding the  $K$  shortest paths between pairs of nodes [20]. The path between any pair of nodes can be represented as a 3D matrix, where the planes are actually 2D matrices that represent the constitutive links on the path represented by the 3D matrix. If the traffic request is represented as a rectangle, then the spectrum allocation comes down to simple search for free space in this 3D matrix in which this rectangle can be accommodated. Contiguity and continuity constraints must be met in order to serve traffic requests in an elastic optical network. This means that the request can be served only if the rectangle can be placed at the same position in all planes of the matrix representing the path between the source and the destination.

The sequence after the SERVE\_FRI state symbol describes the algorithm operation while serving the priority real time traffic (category 1). This is shown in Figure 3, using a flowchart.

For each FRI request, the set  $P$  containing  $K$  shortest paths from the source  $s$  to the destination  $d$  is found using [20]. In Equation (4),  $P$  denotes the set of paths corresponding to one request  $f(s, d, t_a, n_{rf_s})$ , and  $p_{sd}$  denotes the  $k^{\text{th}}$  path from the source  $s$  to the destination  $d$ ,  $1 \leq k \leq K$ .

$$P = \bigcup_{(s,d) \in J^2} p_{sd} \quad (4)$$

In order to serve the request, the controller is searching for free resources among the first path ( $k=1$ ), i.e. the controller is searching for free frequency slots blocks ( $B_k = \{f_{ij} \mid f_{ij} = [f_i, f_j], n_{rfs} \leq f_j - f_i\}$ ). Next, the controller checks whether there are available resources along the path that correspond to the request. The first  $n_{rfs}$  frequency slots are then assigned to the connection.

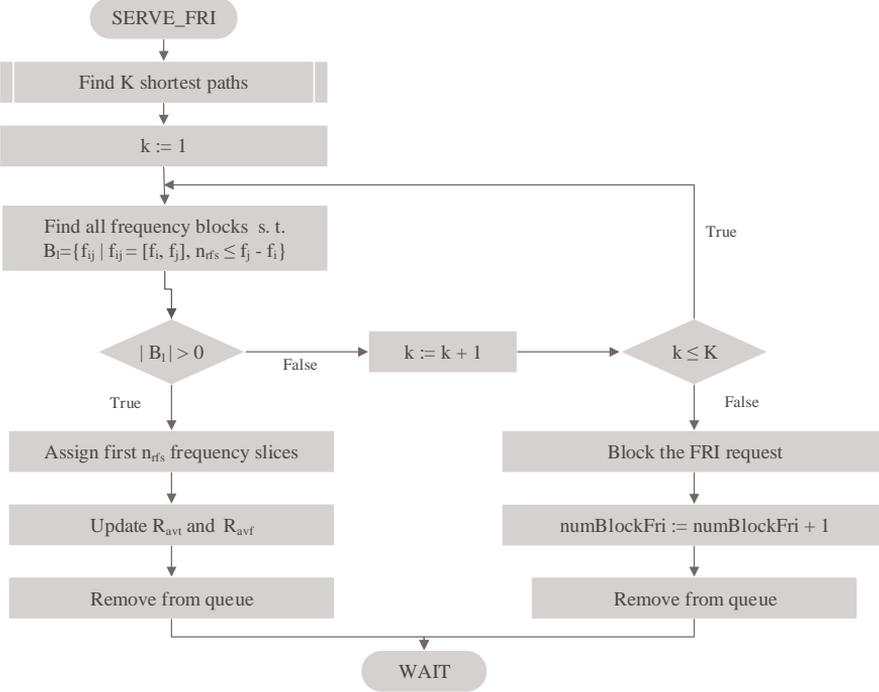


Figure 3

A flowchart that describes the operation of the algorithm, while serving a priority traffic request

Once the FRI request is served, the SDN controller makes a record about the amount of network resources allocated to this request. The average number of frequency slices assigned to the FRI connections ( $R_{avf}$ ) is calculated using Equation (5). The average number of frequency slots assigned to the FRI connections is calculated after each served FRI request and it is referring to the last  $N_F$  served requests.

$$R_{avf} = \frac{1}{N_F} \sum_{i=1}^{N_F} L_{F,i} \quad (5)$$

The average duration of the FRI connections ( $R_{avt}$ ) is calculated using the Equation (6). The average duration of the FRI connections is calculated after each served FRI request and it is referring to the latest  $N_F$  requirements.

$$R_{avt} = \frac{1}{N_F} \sum_{i=1}^{N_F} T_{F,i} \quad (6)$$

Once the  $R_{avf}$  and  $R_{avt}$  values are updated, this request is removed from the queue and the process goes to the *WAIT* state where it is waiting for the next request.

If free resources cannot be found along the first path, then the same check is made for the next path. This procedure is repeated as long as  $k \leq K$  or until appropriate block of free resources is found. If the controller could not find a block of available resources along all  $K$  paths that belong to the set  $P$  for the request to be served, a record is made that this request is blocked, the number of blocked *FRI* requests is increased by one, this request is removed from the queue and the process goes to the state *WAIT* where it is waiting for a new request to arrive.

The sequence after the *SERVE\_BDT* state symbol describes the algorithm behavior while serving big data traffic request (category 2). This is illustrated using flowchart in Figure 4.

Similar to the serving of *FLRs* algorithm, for each *BDT* request described as  $b(s, d, t_s, t_e, n_{rb})$ , the  $K$  shortest paths are calculated using [20].

Only a fixed part of the available network resources is allocated to the *BDT* requests in regards to the priority category 1 by the *SDN* controller. The maximum number of frequency slices that can be assigned to a *BDT* request ( $R_f$ ) is part of the average number of frequency slices assigned to the served *FRI* connections ( $R_{avf}$ ). The size of this part depends on the coefficient  $q_{freq}$ . Similarly, the number of time slots that can be occupied by a *BDT* request ( $R_t$ ) is part of the average duration of the served *FRI* connections ( $R_{avt}$ ) and the size of this part depends on the coefficient  $q_{time}$ . When a *BDT* request arrives in the network, the *SDN* controller calculates  $R_f$  and  $R_t$  using Equations (7) and (8).

$$R_f = \lceil q_{freq} \times R_{avf} \rceil \quad (7)$$

$$R_t = \lceil q_{time} \times R_{avt} \rceil \quad (8)$$

The *BDT* request will be blocked if, and only if, it arrives after the maximum tolerable delay ( $t_s \geq t_e$ ). Then, the controlling process makes a record that this request was blocked, the number of blocked *BDT* requests is increased by one, the request is removed from the queue, and the process goes to a *WAIT* state, waiting a new request to arrive at the network, as can be seen in Figure 4.

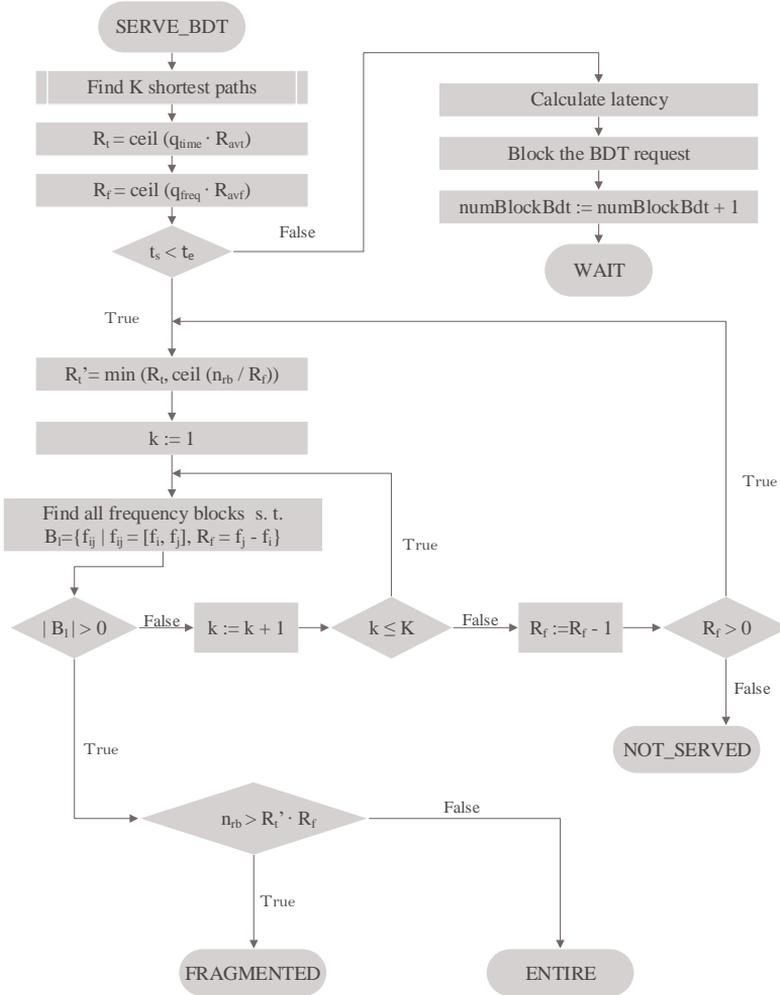


Figure 4

A flowchart that describes the operation of the algorithm, while serving a big data traffic

The network controller tries to assign a maximum possible number of frequency slices to the BDT request that requires  $n_{rb}$  resource blocks. If less than  $R_t$  time slots are needed for serving  $n_{rb}$  resource blocks using  $R_f$  frequency slices, a lower and appropriate time is assigned to the request. The number of time slots that will be assigned to the BDT request ( $R'_t$ ) is calculated using Equation (9).

$$R'_t = \min\left(R_t, \left\lfloor \frac{n_{rb}}{R_f} \right\rfloor\right) \quad (9)$$

In order the request to be served, the controller is searching for free resources along the first path ( $k=1$ ), i.e. the controller is searching for free frequency slots blocks ( $B_l = \{f_{ij} \mid f_{ij} = [f_i, f_j], n_{rfs} \leq f_j - f_i\}$ ). Next, the controller is checking whether there are available resources along the path that correspond to the request. This is repeated until  $R_f > 0$ . If a block of free frequency slices is found ( $|B_l| = 0$ ), the controller checks if  $R_f$  frequency slices and  $R'_t$  time slots are enough for the request to be served entirely. If  $n_{rb} > R'_t \cdot R_f$ , then the process goes to the state *FRAGMENTED*, otherwise the process goes in the state *ENTIRE*. If  $R_f = 0$  and there is no suitable block of free frequency slots that can be used for serving the request, the process goes to the state *NOT\_SERVED*.

The frequency slices with smallest indices are always assigned to the connection. If in the first path, there is no appropriate block of frequency slices for the request to be served ( $|B_l| = 0$ ), the same search is repeated for the next path. This procedure is repeated as long as  $k \leq K$ . Finally, if all  $K$  paths from the set  $P$  are searched and no appropriate resources are found, the controller is trying to serve the request with a smaller number of frequency slices. Namely, the value of  $R_f$  is decreased by one. For the new value of  $R_f$ , a new value for  $R'_t$  is calculated and again all  $K$  paths are searched for suitable block of frequency slices (corresponding to the new values of  $R_f$  and  $R'_t$ ).

If the process is in the state *FRAGMENTED*, the number of resource blocks that can be assigned to the request is lower than the number of resource blocks it requires. In that case, the request will be partially served as it is depicted in Figure 5 (a). If a BDT request is served partially, it is transformed into a new request and its arriving moment will change according to Equation (10).

$$t_s \leftarrow t_s + R'_t + t_{backoff} \quad (10)$$

Also, the number of resource blocks it requires will be updated according to the Equation (11). The number of required RBs will be equal to the initial number of required RBs minus the number of served RBs. The number of served RBs is equal to the product of time spent for serving the block and the number of frequency slots assigned to the *BDT* request. After that, the process goes in the state *WAIT* where it is waiting for a new request to arrive in the network.

$$n_{rb} \leftarrow n_{rb} - (R'_t \cdot R_f) \quad (11)$$

When all the fragments of one request are handled, it is considered to be served.

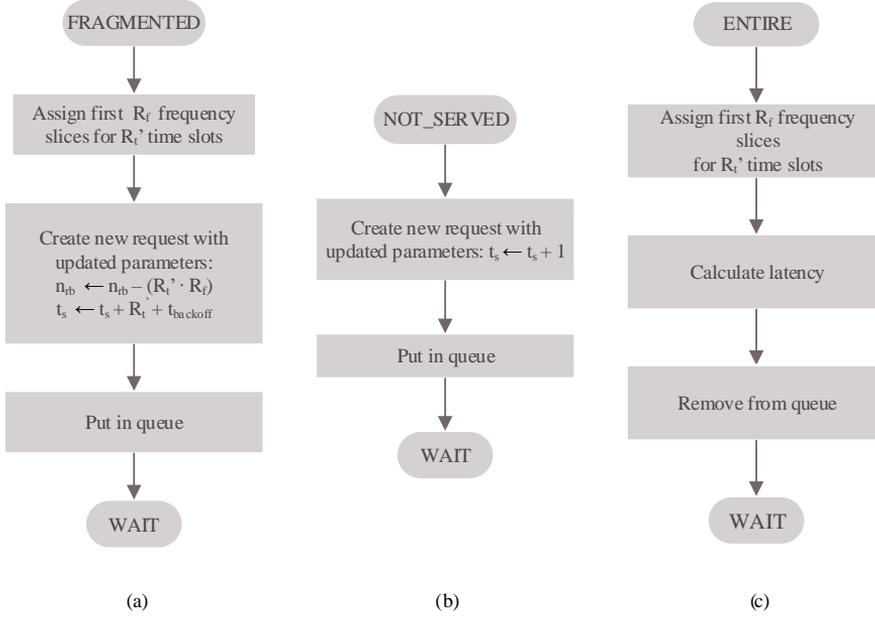


Figure 5

A flowchart that describes the operation of the algorithm while serving a big data traffic (a) BDT request is served partially (b) there are no available free resources for the BDT request to be served (c) BDT request is served entirely

If the controller cannot find free resources for the BDT request among all  $k$  paths for all possible values of  $R_f$ , the BDT request is not blocked but it is put back in the queue with new arrival moment:

$$t_s = t_s + t_{backoff} \quad (12)$$

Once the request arrival time is updated, the process returns to the state *WAIT* where it waits for a new request to arrive in the network as shown in Figure 5 (b).

If the number of resource blocks that may be assigned to the BDT is greater than or equal to the number of resource blocks it requires, an attempt is made for serving the request without fragmentation, as shown in Figure 5 (c). In this case,  $R_f$  frequency slices are assigned to the request for  $R_t$  time slots and the latency is calculated. If this request is created as part of any previous request that could not be fully served, the latency is calculated starting from the moment when the initial request arrived in the network. Once the request is served, it can be removed from the queue and the process goes back to the state *WAIT*, where it waits for a new request to arrive in the network.

## 5 Results and Discussion

In this section, the simulation results are presented for the purpose of analysis, evaluation and validation of scheduling scheme for spectrum assignment in elastic optical networks. For simplicity, we use  $k=1$ .

Figure 6 shows the average latency of the BDT traffic as a function of  $q_{freq}$ . It is obvious that latency decreases for greater values of the parameter  $q_{freq}$ . As mentioned, a greater value of  $q_{freq}$  allows for larger portion of the spectrum resources to be available for serving BDT requests. Thus, BDT requests are served with less fragmentation that directly reduces the latency. In other words, if  $q_{freq}$  has a small value, the request needs to be fragmented several times into smaller blocks, before it can be served, resulting in greater latency or, in the extreme, breach the maximum tolerable delay. According to Figure 6, as the BDT load increases the latency becomes slightly higher. Obviously, heavy BDT loads congest the network and there are less free resources, but the average BDT latency is not affected significantly.

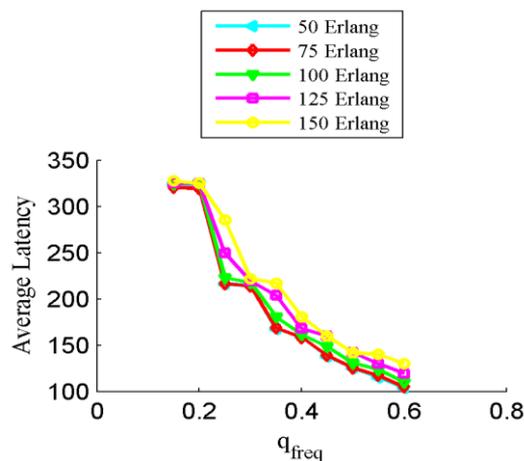


Figure 6

Average latency (in TS) as a function of  $q_{freq}$  for different BDT traffic loads

Figure 7 plots the FRI blocking probability versus  $q_{freq}$  for different BDT traffic loads. As can be seen, higher data traffic load causes higher blocking probability for FRIs. On one hand, network operators want higher network utilization, while aiming smaller blocking probability for the critical services. Clearly, a tradeoff must be considered by the carriers. Figure 7 further shows that the blocking probability of FRIs can be slightly controlled by the change of design parameter  $q_{freq}$ . The change of  $q_{freq}$  has not such influence over the FRI blocking probability, compared to the BDT load influence.

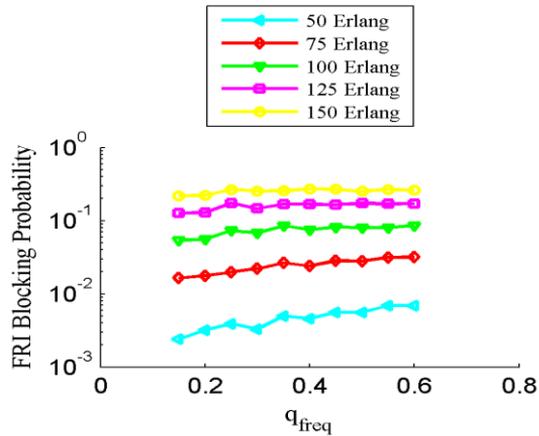


Figure 7

FRI blocking probability as a function of  $q_{freq}$  for different BDT traffic loads

BDT requests tend to reserve spectrum resources in advance. Therefore, the current resources available for FRI requests are wasted. This behavior leads to higher blocking probability for FRIs compared with BDT, as can be seen from Figures 7 and 8. In Figure 8 the BDT blocking probability is plotted as a function of  $q_{freq}$ . The blocking probability decreases as the value of  $q_{freq}$  increases. When  $q_{freq}$  is larger, there are more resources available for the arriving BDT requests, and thus the probability that the request will not be served before the maximum tolerable delay expires, becomes low.

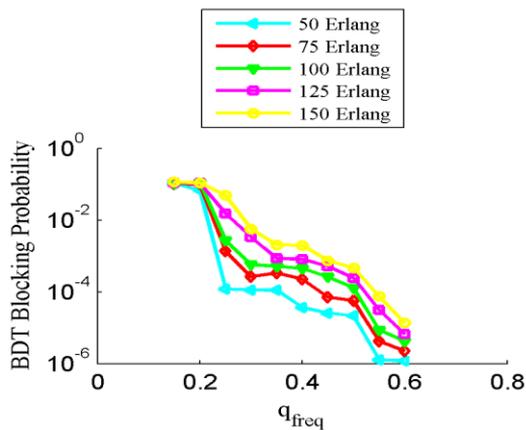


Figure 8

BDT blocking probability as a function of  $q_{freq}$  for different BDT traffic loads

Figure 9 shows the effect of the coefficient  $q_{freq}$  and the BDT traffic load on the network utilization defined as a percentage of the time-frequency slots that are successfully used for carrying traffic (FRI or BDT). It can be seen that the average network utilization does not change with respect to the coefficient  $q_{freq}$ , while it is significantly influenced by the volume of the BDT traffic. The results reveal higher network utilization, for the case of higher BDT traffic loads. Thus, it can be concluded that for fixed FRI traffic loads, adding more BDT traffic will cause increased network utilization.

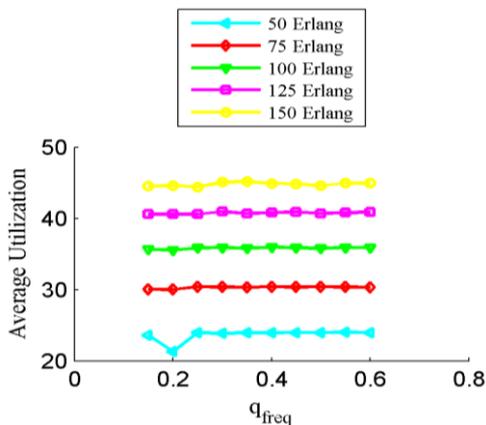


Figure 9

Average network utilization (in %) as a function of  $q_{freq}$  for different BDT traffic loads

Based on these results, we can summarize that this paper enhances the previous research, by proposing an improved RSA method for parallel handling of data traffic in EON, as well as, investigating the average network latency and utilization in case of multiple traffic categories. These findings are important for efficient coexistence of different classes of services. The obtained results have direct practical relevance. Namely, a network operator can improve the overall network conditions choosing appropriate values for the controlling coefficients, as well as, data traffic loads.

## Conclusions

RSA is one of the most intriguing research topics in the field of flexible optical networks. Designing an algorithm that efficiently deals with the traffic requests in an EON, can significantly improve the performance of the network. This is particularly challenging in the case of a network, in which, different categories of traffic coexist.

This paper presents a simulation tool, along with a formal description of the controlling algorithm, in SDN-based EON. In this work new performance metrics are introduced and the impact of the controlling algorithm's coefficients on the network conditions is analyzed. In the simulations, we highlighted the coefficient

that determines the portion of the frequency slots that can be assigned to BDT requests. From the obtained results, it can be concluded that this coefficient significantly impacts the BDT blocking probability and BDT latency, while it has small influence over the FRI blocking probability and network utilization. This proves that through  $q_{freq}$  the network operator can actively control the BDT performance, with a possibility to introduce different sub-classes of service.

Conversely, the results revealed that the BDT traffic load has a significant impact over the blocking probability (for both FRI and BDT requests), and it is a crucial parameter to control the overall network utilization.

Further research of this issue will target different RSA algorithms, in order to find more efficient spectrum management schemes, which can reduce the blocking probability and improve the network capacity utilization. The future work will involve optimization of the controlling coefficients calculation, as well as, analysis of the influence of the suggested algorithm on the network fragmentation. Clearly, more study is needed to find an appropriate approach for reducing the FRI traffic degradation at high intensity of BDT traffic. One possible solution may be to consider a smaller number of possible paths while serving the BDT traffic. This raises the question of how many paths must be considered while searching for free resources for the first and second traffic category, as well as, how the decision will be made for which of the  $K$ -shortest paths will be considered and why.

## References

- [1] S. Shahsavari, H. Beyranvand, and J. A. Salehi, "Multi-quality of service routing and spectrum assignment in elastic optical networks", IEEE ICC Optical Networks and Systems Symposium, pp. 1-6, Paris, France, May 2017
- [2] Y. Xiong, Y. Li, X. Dong, Y. Gao, and G. N. Rouskas, "Exploiting SDN principles for extremely fast restoration in elastic optical datacenter networks", IEEE Global Communications Conference, Washington, DC, USA, February 2017
- [3] Y. Xiong, Y. Li, B. Zhou, R. Wang, and G. N. Rouskas, "SDN enabled restoration with triggered precomputation in elastic optical inter-datacenter networks", J. Opt. Commun. Netw., Vol. 14, No. 1, pp. 24-34, January 2018
- [4] D. Amar, E. Le Rouzic, N. Brochier, and C. Lepers, "Class-of-service-based multilayer architecture for traffic restoration in elastic optical networks", J. Opt. Commun. Netw., Vol. 8, No. 7, pp. 34-44, July 2016
- [5] H. Xuan, Y. Wang, Sh. Guan, and Z. Xu, "A new optimization model and algorithm for a network scheduling problem in inter-datacenters elastic optical networks", International Conference on Computational Intelligence and Security, Hong Kong, December 2017

- [6] W. Fang, M. Zeng, X. Liu, W. Lu, and Z. Zhu, "Joint spectrum and it resource allocation for efficient vnf service chaining in inter-datacenter elastic optical networks," *IEEE Communications Letters*, Vol. 20, No. 8, pp. 1539-1542, 2016
- [7] A. V. S. Xavier, R. C. Almeida Jr., J. F. Martins-Filho, D. A. R. Chaves, and C. J. A. Bastos-Filho, "Spectrum contiguity and continuity based dedicated protection for flexible optical networks", *Journal of Microwaves, Optoelectronics and Electromagnetic Applications*, Vol. 16, No. 2, pp. 481-493, June 2016
- [8] L. Velasco, M. Ruiz, K. Christodouloupoulos, M. Varvarigos, M. Żotkiewicz, and M. Pióro, "Routing and spectrum allocation", in *Elastic optical networks: architectures, technologies, and control*, Springer, Switzerland, pp. 55-80, 2016
- [9] B. C. Chatterjee, N. Sarma, and E. Oki, "Routing and spectrum allocation in elastic optical networks: A tutorial", *IEEE Communication Surveys and Tutorials*, Vol. 17, No. 3, pp. 1776-1799, 2015
- [10] R. Goéscie, K. Walkowiak, and M. Klinkowski, "On the complexity of routing and spectrum allocation in survivable elastic optical network with unicast and anycast traffic", *8<sup>th</sup> International Workshop on Resilient Networks Design and Modeling*, pp. 166-173, Halmstad, Sweden, September 2016
- [11] P. Afsharlar, A. Deylamsalehi, J. M. Plante, J. Zhao, and V. M. Vokkarane, "Routing and spectrum assignment with delayed allocation in elastic optical networks", *J. Opt. Commun. Netw.*, Vol. 9, No. 3, pp. 101-111, March 2017
- [12] J. Wu, M. Xu, S. Subramaniam, H. Hasegawa, "Routing, fiber, band, and spectrum assignment (RFBSA) for multi-granular elastic optical networks", *IEEE ICC 2017 Optical Networks and Systems Symposium*, Paris, France, May 2017
- [13] W. Jia, Z. Xu, Z. Ding, and K. Wang, "Routing and spectrum assignment algorithm with prediction for elastic optical networks under self-similar traffic", *15<sup>th</sup> International Conference on Optical Communications and Networks*, pp. 1-3, Hangzhou, China, September 2016
- [14] C. L. Triveni, P. C. Srikanth, and T. Srinivas, "An optimized routing algorithm for elastic optical network", *International Conference on Electrical, Electronics, and Optimization Techniques*, pp. 3873-3878, Chennai, India, March 2016
- [15] M. Klinkowski and K. Walkowiak, "A heuristic algorithm for routing, spectrum, transceiver and regeneration allocation problem in elastic optical networks", *18<sup>th</sup> International Conference on Transparent Optical Networks*, pp. 1-4, Trento, Italy, July 2016

- [16] Q. P. Van et al., “Virtualized routing and frequency allocation functions in elastic optical networks”, 42<sup>nd</sup> European Conference and Exhibition on Optical Communications, pp. 1-3, Dusseldorf, Germany, September 2016
- [17] H. Zheng and H. T. Mouftah, “Supporting advance reservations in wavelength-routed WDM networks,” in IEEE Int. Conf. on Computer Communications and Networks, pp. 594-597, October 2001
- [18] S. Sugihara, Y. Hirota, S. Fujii, H. Tode, and T. Watanabe, “Dynamic resource allocation for immediate and advance reservation in space-division-multiplexing-based elastic optical networks”, J. Opt. Commun. Netw., Vol. 9, No. 3, pp. 183-197, March 2017
- [19] National Science Foundation, “A partnership for high - speed networking: final report 1987-1995,” pp. 20-39, Available: <https://web.archive.org/web/20170202190225/http://nsfnets-legacy.org/about.php>
- [20] J. Y. Yen, “Finding the k shortest loopless paths in a network,” Management Science, Vol. 17, No. 11, pp. 712-716, 1971