Ireneusz OLSZEWSKI

Uniwersytet Technologiczno-Przyrodniczy, Instytut Telekomunikacji i Informatyki

Algorithms of Routing and Spectrum Assignment in Spectrum Flexible Transparent Optical Networks

***Abstract**.. The considered problem of routing and spectrum assignment (RSA) takes into consideration minimizing the length of the transmission distance under spectrum continuity constraints and the relationship between the traffic bitrates and the spectrum bandwidth. The proposed algorithms constitute modifications of the well-known Dijkstra's algorithm. The obtained results show that a significant reduction in the number of rejected requests can be achieved by an appropriate selection of the scheme spectrum segments in the aggregated spectrum of the path but not by increasing the computational complexity of the algorithms.*

***Streszczenie**. Rozważany problem rutingu I przypisania widma (RSA) obejmuje minimalizację długości ścieżki transmisyjnej przy warunkach nałożonych na ciągłość widma oraz zależność pomiędzy szybkością binarną i widmem pasma sygnału. Zaproponowane algorytmy stanowią modyfikację algorytmu Dijkstry. Uzyskane wyniki pokazują, że istotna redukcja liczby odrzuconych żądań może być uzyskana nie poprzez zwiększanie złożoności obliczeniowej algorytmu, ale poprzez odpowiedni schemat wyboru segmentu widma z zagregowanego widma ścieżki. **(Algorytmy rutingu i przydziału widma w elastycznych przezroczystych sieciach optycznych)**.*

Keywords: routing and spectrum assignment (RSA) problem, lightpaths, optical networks, OFDM modulation.
Słowa kluczowe: problem rutingu i przypisania widma, ścieżka optyczna, sieć optyczna, modulacja OFDM.

Introduction

A spectrum-sliced elastic optical path network (SLICE) has been proposed as an efficient solution for a flexible bandwidth allocation in optical networks. The used optical orthogonal frequency-division multiplexing (OFDM) supports the transport of multi-granularity Internet traffic, however, requires a grid-flexible (sliced or mini-grid) or a fully gridless network. In SLICE, spectrums of OFDM, signals are flexible and sliced to utilize arbitrary contiguous spectrum slots transferring traffic with arbitrary bitrates. Similarly to Routing and Wavelength Assignment in DWDM optical networks, Routing and Spectrum Assignment (RSA) problem appears in the SLICE networks. In [1] a dynamic RSA problem, which takes into account the relationship between the spectrum bandwidth, signal format and traffic bitrates has been formulated. The objective function includes minimizing the path length, and spectrum continuity constraints and spectrums of OFDM signals non-overlap constraints play the role of constraints in the formulated optimization problem. In addition, two different algorithms for solving the formulated problem have been proposed, based on the spectrum segment representation, which naturally supports both mini-grid and fully gridless networks. On the other hand, in [2] a dynamic routing algorithm with distance adaptive modulation in SLICE networks, has been proposed. In [3] static RSA problem is formulated as an integer linear programming problem and a heuristic algorithm solving this problem has been proposed if the ILP solution is not feasible. In [4] *k*-path Signalling RSA-based scheme has been proposed and a simulation of results shows that it works better than other RSA schemes in Bandwidth Flexible Optical Networks. In [5] the problem of optical network planning based on OFDM, in which the connections are protected by spectrum non-overlap rule, has been formulated. To solve this RSA problem, several algorithms ranging from optimal and decomposition ILP algorithms to a sequential heuristic algorithm combined with appropriate ordering policies and simulated annealing meta-heuristic, have been proposed. In [6] the problem of serving dynamic traffic in a spectrum flexible optical network is considered, where the spectrum allocated to an end-to-end connection varies dynamically with time so as to follow the required source transmission rate.

It should be noted that further studies of the RSA problem are needed to propose new routing algorithms minimizing the blocking probability of requests with a possibly small function of computational complexity.

In this paper, two algorithms for solving the basic RSA problem that are based on the well-known Dijkstra's algorithm have been proposed. The proposed scheme selection of the segments from the available spectrum of the path provides significant reduction in the number of rejected requests as compared to other known algorithms.

The remaining part of this paper is organized as follows. The first part describes the formulation of the optimization problem. The second part gives an overview of the compared algorithms and proposes two heuristic algorithms to solve this problem. The third part contains the results of simulation, obtained after application of the presented algorithms. The last section presents the final conclusions.

Formulation of optimization problem

Let *G (V, E, D)* be the network, where *V* is the set of nodes, *E* is the set of unidirectional links and *D* is the set of lengths of links. *R* is the symbol rate for each subcarrier (in baud), while *G* (Hz) is a guard band between adjacent OFDM signals. In this paper, as in [1], segment representation of the spectrum is assumed that supports coarse WDM grid networks, mini - grid (slot based) networks and fully grigless networks. Therefore, the spectrum S_{ij} of link *(i, j)* can be defined as [1]:

$$(1)\qquad S_{i,j} = \bigcup_{k=1}^{K_{ij}} S_{ij}^{k} = \bigcup_{k=1}^{K_{ij}} (a_{ij}^{k}, b_{ij}^{k}) \subseteq (f^{start}, f^{end})$$

where: $S_{ij}^{k} = (a_{ij}^{k}, b_{ij}^{k})$ is *k* segment of an available contiguous spectrum of link *(i, j)*, such that $a_{ij}^{k} \prec b_{ij}^{k} \le a_{ij}^{k+1} \prec b_{ij}^{k+1}, k \in (1,2,..., K_{ij})$ and K_{ij} is the number of available segments of this link. $B=(f^{start}, f^{end})$ is the spectral window bandwidth of each link *(i,j)*. Furthermore, let the current request between a pair of nodes *s, d*$\in$*N* be for C units of bandwidth [in bps]. The relationship between the traffic bitrate *C* and the spectrum of the signal *B* when using OFDM modulation can be defined as follows [1]: $B = (\lceil C/2mR \rceil + 1)R$, where *m* is the number of bits per symbol and *R* is the symbol rate (in baud) for each subcarrier. The above dependence was obtained basing on the assumption that all the subcarriers have the same signal format and use PDM and AMF [1]. Before formulation of the optimization problem the variables used in this formulation need to be defined: A^{sd} - binary variable equals 1 if the request is accepted and 0 otherwise, r_{ij} – binary variable equals 1 if the OFDM lightpath passes through link *(i, j)* and 0 otherwise, r_{ij}^{k} - binary variable equals 1 if the OFDM signal passes through link *(i, j)* and the spectrum of the signal is included in the *k*-th segment $S_{ij}^{k} = (a_{ij}^{k}, b_{ij}^{k})$. Variables f_a, f_b define a contiguous range of the signal spectrum for the incoming request, i.e. (f_b - f_a)=$B+G$. Generally speaking, the considered problem is to find the shortest path P_{sd} between pair of nodes (*s, d*), for which aggregated spectrum S_{sd} enables realization of the incoming request. Aggregated spectrum of path S_{sd} should be understood as spectrum intersection of links belonging to the path, i.e. $S_{s,d} = \bigcap_{(i,j)\in P_{s,d}} S_{ij}$, where S_{ij} is the spectrum of link *(i,j)*. A formulation of the complete dynamic RSA problem with the constraints of the spectrum continuity and the constraints of the format transmitted signal constraints has been shown below [1]:

$$Max\left[\alpha A^{sd} - \sum_{(i,j)\in E}(r_{ij}D_{ij})\right]; \qquad \alpha > \sum_{(i,j)\in E} D_{ij} \tag{2}$$

The objective function (2) minimizes the length of the path being selected.

$$\sum_{j} r_{ij} \le A^{sd} \qquad \forall_{i \in N} \tag{3}$$

Constraint (3) ensures that the selected path does not contain cycles, which means that for each node *i,* on the path being chosen between a pair of nodes *(s, d)*, ($A_{sd} = 1$), only one arc *(i, j)* belongs to the path ($r_{ij} = 1$).

$$\sum_{i} r_{ij} - \sum_{i} r_{ji} = \begin{cases} -A^{sd}, & j = s \\ A^{sd}, & j = d \\ 0, & j \ne s, d \end{cases} \qquad \forall_{i \in N} \tag{4}$$

Equations (4) give the flow balance for the path.

$$\sum_{k=1}^{K_{ij}} r_{ij}^{k} = r_{ij} \qquad \forall_{ij \in N} \tag{5}$$

Equations (5) ensure that only one available segment S_{ij}^{k} is selected on link *(i, j)* of the path being chosen between a pair of nodes *(s, d)*.

$$f_b - f_a - G = B = \left(\lceil C/2mR \rceil + 1\right)R \tag{6}$$

Equation (6) defines the relationship between the spectrum bandwidth *B* and bitrate C for the incoming request.

$$\begin{cases} f_b - b_{ij}^{k} \le B^{total}\left(2 - r_{ij} - r_{ij}^{k}\right) \\ a_{ij}^{k} - f_a \le B^{total}\left(2 - r_{ij} - r_{ij}^{k}\right) \end{cases} \qquad \forall ij \in E, \forall k \tag{7}$$

Equations (7) ensure the spectrum non-overlap rule. $B^{total} = f^{end} - f^{start}$ is the total spectrum bandwidth of each link *(i, j)*. If the OFDM signal path of the current request passes through link *(i, j)* and is implemented within the *k*-th segment S_{ij}^{k} (r_{ij}=1, r_{ij}^{k}=1), then the system of inequalities (7) becomes: $a_{ij}^{k} \le f_a$, $f_b \le b_{ij}^{k}$ which means that the signal spectrum is located in the segment. However, for the other segments $S_{ij}^{\bar{k}}$, $\bar{k} \in \{1, 2, ..., k-1, k+1, ..., K_{ij}\}$, that have not been changed, (r_{ij}=1, r_{ij}^{k}=0) the system of inequality becomes: $f_b - b_{ij}^{\bar{k}} \le B^{total}$, $a_{ij}^{\bar{k}} - f_a \le B^{total}$ which means that the spectrum of the signal does not exceed the total bandwidth of the link.

$$\sum_{ij \in E} r_{ij} D_{ij} \le TD_{\max}(m) = TD_{\max}(1)/2^{m-1} \tag{8}$$

Inequality (8) is the restriction on the transmission distance. The maximum value of the transmission distance TD_{max} depends on the used OFDM system and decreases by half with the increase in the number of bits *m* per symbol (accordance with the half-distance law in [7]).

Because of non-linear constraints (6) and (8), the presented complete dynamic RSA problem is an integer non-linear programming problem.

Heuristic algorithms-sub-optimal solution

In [1] a decomposition method for solving the complete dynamic RSA problem in three steps have been developed: in the first of them, the signal format selection and determination of the value of m are performed. In the second step, the primary RSA problem, defined by equations (2) - (7) is solved. Solving this problem is based on determining the shortest path with adequate spectrum assignment for the incoming request. The third step includes checking condition (8).

In [1] three different algorithms have been proposed to solve the basic RSA problem. The first of these, marked as SPV (Spectrum - Vector Constraint Path Searching Algorithm), generates a vector path searching tree (PVST), which is similar to the trees generated by algorithms based on the branch-and-bound method. In this tree, the candidate paths are stored together with the aggregated spectrum and tested at each level of the tree. The computational complexity of the algorithm (determined for the worst case) is an exponential function, and is $O(N) = q_m^{|N|-1}$, where q_m is the maximum out-degree of the network. Due to the very limited scalability of the algorithm, a solution generated by the algorithm may be used only to verify the solutions generated by other algorithms for the networks with small numbers of nodes.

The second of the proposed algorithms denoted as KSP (k Shortest Paths) is based on a set - *k* shortest paths, sequenced by the transmission distance (TD), which are calculated off-line for each pair of nodes. For an incoming request between a given pair of nodes the aggregated spectrum is determined for each of *k* designated paths between them. It should be noted that the constant *k*-element set of paths, determined statically, can provide only a sub-optimal solution.

The third algorithm, MSP (Modified Shortest Path Algorithm) is a modification of the well-known Dijkstra's algorithm. Modification of this algorithm is based on the introduction of the spectrum intersection of the links belonging to the paths being chosen and checking if there is a bandwidth available for the incoming request. The computational complexity of this algorithm is polynomial and is $O(N) = K N^2$, where *K* is equal to the average number of segments in the aggregate spectrum of the path. It should be noted that this algorithm can not find a solution, even if there are the bandwidth resources available for the incoming requests. The reason of this is the trap that is based on the fact that the algorithm, after reaching the node on the shortest path, cannot find the link outgoing to the next node or the end node with spectrum bandwidth to allow admission request.

It should be emphasized that in all of these algorithms it is the first feasible segment from the aggregated spectrum of the path, which enables realization of the bandwidth *(B + G)* that is selected. Assuming that the average number of segments in the aggregated spectrum of the path is *K*, the computational complexity of the function detecting the first feasible segment is *O(K)*.

In this paper, next two algorithms for solving the basic RSA problem, based on Dijkstra's algorithm, have been proposed. The first of them is marked as the MSP2 and the second as MSP3.

MSP2 Algorithm

For a given matrix *D* whose elements D_{ij} determine the length of links *(i,j)* and the available spectrum S_{ij} on these links the shortest path is determined basing on a modified Dijkstra's algorithm. Modification of this algorithm is based on introduction of the spectrum intersection of the links belonging to the path being chosen and detects whether the aggregated spectrum enables realization of the request. The general idea of Dijkstra's algorithm [8] is based on the movement on the network arcs, in subsequent iterations, from source node *s* to destination node *d* and marking the intermediate nodes by their current distances from node *s.* The feature of node *u* is fixed when it is equal to the length of the shortest path from node *s* to *u.* During initialization of the algorithm source node *s* receives a fixed feature. Then, in the first iteration, a temporary feature of each successor *v* of node *s* is changed from infinity to the feature equal to weight of arc D_{sv}. The node with the smallest feature of a temporary node, for example node *u*, is replaced by a fixed feature, which does not change until the end of the work of the algorithm. In the next iteration the successors of node *u* are featured. Then, as before, the node with the smallest temporary feature of all, receives a fixed feature. The algorithm terminates when the destination node *d* receives a fixed feature. In each iteration of the algorithm the value of the temporary features is reduced. Let *dist* be an *n*-element vector, where element *dist(v)* is the distance from the source node *s* to node *v* and *pred* is a vector of the predecessors on the shortest path from node *s* to node *d.* Furthermore, let the variable ***newlabel*** be the value of feature of temporary node *v*, determined from node *u* for which the feature has recently been established, i.e. ***newlabel←dist(u)***+ $D_{u,v}$. If the value of feature of node *v*, i.e. the distance from node *s* to node *v* through node *u* is reduced, then *dist(v) ← newlabel* and *pred(v) ←u*. At this point, the algorithm should be modified. Let S_{su} be the aggregated spectrum of the links belonging to the path from node *s* to node *u* and let S_{uv} be the spectrum of the link from node *u* to node *v*. Let the value of the variable $S_{sv} = S_{su} \bigcap S_{uv}$ be determined together with the value of the variable *newlabel.* If the feature node *v* is reduced, i.e. the path from node *s* to node *v* by node *u* is reduced and spectrum S_{sv} enables the realization of the bandwidth *B+G* then the process is continued until the node *d*. Otherwise, the request is blocked. The key of the proposed algorithm is to select the smallest feasible segment $(a_{sd}^k, b_{sd}^k)\ k \in \{1, 2, ..., k, ..., K_{sd}\}$ from the aggregated spectrum of the path, which enables realization of bandwidth *(B + G)*. In order to explain the validity of the proposed scheme, let be assumed that the spectrum of links S_{ij} between pair of nodes *i* and *j*, which limits the aggregated spectrum of the path S_{sd}, has two segments of the available spectrum S_{ij}*=(0-100)∪(150-200) GHz*. If between the pair of nodes *s-d*, or other, the request of bandwidth realization *B+G =50 GHz* appears and link *(i,j)* will belong to the shortest path, the second segment will be occupied. Then, if the link *(i, j)* will be belong to the shortest path between the same or another pair of nodes between which a connection with the bandwidth *B + G = 80* GHz should be set the first segment will be occupied. Otherwise, i.e., if the first segment would be occupied for the first request the second request would be rejected. In the SPV, KSP i MSP [1] the first feasible segment (first-fit selection scheme) that allows the implementation of bandwidth *(B + G)* is always occupied. The algorithm MSP2 minimizing the length of path between a pair of nodes *(s, d)*, which enables the realization of bandwidth *B+G* is shown below. Let *F(*S_{sv}*, B+D)* be a boolean function which that is true if the aggregated spectrum S_{sv} enables the realization of bandwidth *B+G* and *false* otherwise. Let *final* be an *n*-element vector of boolean, where *final(i)* changes its state from *false* to *true* when feature node *i* is changed from temporary to fixed.

Input: request between a pair of nodes *s, d* with bandwidth *(B + G)*
Output: the shortest path with aggregated spectrum S_{sd} which enables realization of bandwidth *(*f_a*-*f_b*)=B+G*
MSP2 Algorithm

for $v \in V$ **do** *dist(v) ←∞, final(v) ←false* **end for**
    *dist(s) ←0, final(s) ←true, u←*s, $S_{ss} \leftarrow (f^{start}, f^{end})$,
while *final(d) = false* **do**
    for each successor *v* of node *u* if not *final(v)* **do**
        newlabel←dist(u)+$D_{u,v}$
        $S_{sv} \leftarrow S_{su} \cap S_{u,v}$; {bandwidth aggregation}
        if (*newlabel < dist(v)*) **and** *F(*S_{sv}*, B+D)* **then**
            dist(v) ← newlabel; {change the shortest path to node *v*}
            pred(v) ← u;
        end if
    end for
    find a node *y* with the smallest temporary feature dist(y), different from ∞;
    if dist(y) < ∞ **then**
        final(y) ←true; { *y* receives a fixed feature }
        u←y;
        if *u* = *d* **then**
            select the smallest segment S_{sd}^k, such that $B + G \subseteq (a_{sd}^k, b_{sd}^k)$, from the aggregated spectrum S_{sd}; $f^a \leftarrow a_{sd}^k,\ f^b \leftarrow f^a + (B+G)$
        end if
    else Rejection of the request
    end if
end while

The length of the shortest path is in variable *dist(t)*, (if *dist(t)*= ∞ then the request of path choice is rejected) and the course of the path can be obtained on the basis of *pred.* Computational complexity of the algorithm, without function *F()*, is $O(|N|^2)$. Assuming that the computational complexity of *F()*, equal to the average number of segments in the aggregated spectrum S_{SV}, is equal to O(K), the computational complexity of MSP2 can be defined as $O(K|N|^2)$.

MSP3 Algorithm

If the feature node *dist(v)* is reduced in the algorithms MSP2 and the aggregated spectrum enables realization of bandwidth *B+G,* the process is continued until node *d* is reached. Otherwise, if feature node *dist(v)* is reduced, while aggregate spectrum $S_{sv}=S_{su}\cap S_{uv}$ does not enable realization of bandwidth *(B+G)* the algorithm falls into the trap too. If even for a longer way, there are free resources for realization of bandwidth *B+G*, the incoming request is rejected. The reason for this is the inability to withdraw the algorithm from node *u*, for which the feature was last fixed. In the MSP3 algorithm, the possibility of withdrawing from node *u* to its predecessor has been realized. After blocking the arc *(pred(u), u)*, by substituting $D_{pred(u),u} \leftarrow \infty$, the withdrawal occurs, i.e. $u \leftarrow pred(u)$. Then, the process of finding the next link from node *u*, which enables realization of the bandwidth on the shortest path, starts from the beginning. The request is rejected if all the outgoing paths from node *s* to node *d* are impossible to be realized with the feasible bandwidth. MSP3 algorithm is shown below. Let *degree(s)* be the out-degree of node *s.*

Input: request between a pair of nodes *s, d* with bandwidth (B + G)
Output: the shortest path with aggregated spectrum S_{sd} which enables realization of bandwidth $(f_a\text{-}f_b)=B+G$;
MSP3 Algorithm

for $v\in V$ **do** $dist(v) \leftarrow \infty$, $final(v) \leftarrow false$ **end for**
$dist(s) \leftarrow 0$, $final(s) \leftarrow true$, $u \leftarrow s$, $S_{ss} \leftarrow (f^{start}, f^{end})$,
while *final(d) = false* **do**
  for each successor *v* of node *u* if not *final(v)* **do**
    if $u = s$ **then** $S_{su} \leftarrow (f^{start}, f^{end})$ **end if**
    $newlabel \leftarrow dist(u)+D_{u,v}$
    $S_{sv} \leftarrow S_{su} \cap S_{u,v}$; { bandwidth aggregation}
    if *(newlabel < dist(v))* **and** $F(S_{sv}, B+D)$ **then**
      $dist(v) \leftarrow newlabel$; { change the shortest path to node *v*}
      $pred(v) \leftarrow u$;
    end if
  end for
  find a node *y* with the smallest temporary feature dist(y), different from ∞
  if dist(y) < ∞ **then**
    $final(y) \leftarrow true$; { *y* receives a fixed feature }
    $u \leftarrow y$;
    if $u = d$ **then**
      select the first segment S_{sd}^{k}, such that
      $B+G \subseteq (a_{sd}^{k}, b_{sd}^{k})$, from the aggregated
      spectrum S_{sd} ; $f^{a} \leftarrow a_{sd}^{k}$, $f^{b} \leftarrow f^{a}+(B+G)$,
      $final(d) \leftarrow true$
    end if
  else if $u \neq s$ **then**
      $D_{pred(u),u} \leftarrow \infty$, $dist(u) \leftarrow \infty$, $u \leftarrow pred(u)$,
      if u=s **then** $degree(s) \leftarrow degree(s)-1$
    else $degree(s) \leftarrow 0$ **end if**
  end if
  if *degree(s) = 0* **then** Rejection of the request,
    $final(d) \leftarrow true$
  end if
end while

The difference between the MSP and the SPV 3 [1] is that in case of MSP3, if there is no outgoing link with an available bandwidth from the node with the last fixed feature, the algorithm withdraws to the previous node and attempts to find the next outgoing link from that node. If there is not any link with an available bandwidth from this node, either, the algorithm withdraws to the previous node, etc. In turn, the SPV algorithm searches all the nodes in the generated tree of solutions that represent the paths from node s to other nodes in the network. From these nodes which define the path from node *s* to node *d* the SPV algorithm selects the shortest path. The SPV algorithm searches the set of all potential solutions as shown above, while the proposed MSP3, searches only a subset of nodes in the tree generated by SPV algorithm and terminates after finding the first path between a pair of nodes *s* and *d.* Therefore, the proposed algorithm is generally faster than SPV. It should be noted, however, that in the worst case, the computational complexity of the MSP3 algorithm is also exponential.

Obtained results

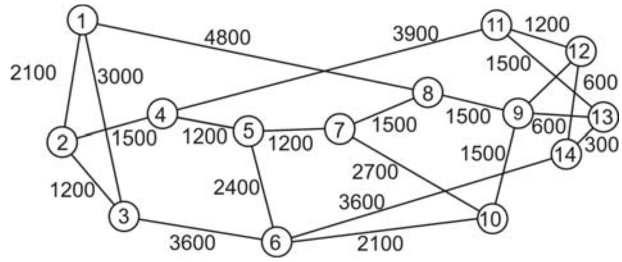

Fig. 1. Topological structure of the network - the NSFNET

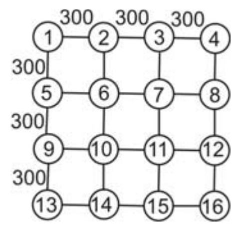

Fig. 2. Topological structure of the network - the GRID

Verification of the algorithms was made for two different networks. The first one, for which the topological structure is shown in Figure 1 [1] contains 14 nodes connected by links with the continuous spectrum equal to B^{total} = *400GHz*. The second [1], for which the topological structure is shown in Figure 2 contains 16 nodes connected by the links with the same continuous spectrum. Each edge of the graph is a pair of the unidirectional links. In this work, it was assumed that each node can be input and the output node. Thus, in the first network, 196 pairs of nodes can be distinguished, while in the second 256 pairs of nodes. On the edges of the graph the length of the links of the network are presented. It is assumed that the stream of requests between each pair of nodes (s, d) is the Poisson with intensity λ, and holding time of the call has an exponential distribution with an average value equal to $1/\mu=1$. The bandwidth of request is uniformly distributed from 10 GBps to 200 Gbps with an average value equal to $\overline{C}$ = 105 Gbps. The volume of traffic (in bps) between each pair of nodes is $\rho\overline{C}$, where ρ is the volume of traffic in the erl. Network simulation was carried out in static and dynamic conditions. In static conditions connections are only set up (long-lived connections), while in dynamic conditions connections are set up and disconnected (short-lived connections). In static conditions for a given trial of the simulation, all algorithms are verified for the same stream of requests. In dynamic conditions the results are recorded after obtaining an equilibrium state of the system. Both in static and dynamic conditions, tests have been done for *T=10* trials. The number of requests in the network is a condition for the end of the simulation. In the dynamic conditions each trial included 25000 requests, while in static conditions each trial included 5000 requests. The confidence intervals were determined for α=0.05. In addition, it was assumed that the systems have the same OFDM modulation rate equal to $R = 1$ Gbaud. Tables 1 ÷ 4 show the number of rejected requests, blocking probability, spectrum utilization ratio and processing time of the algorithms, depending on the amount of traffic offered to the network for a different modulation level $m = 2, 4$. Blocking probability is defined as the capacity blocking probability equal to $\sum C_i A_i / \sum C_i$, where C_i is the capacity of the i-th request while A_i is a binary variable equal to 1 if the request is accepted and 0, otherwise The spectrum efficiency ratio is defined as the ratio of the occupied bandwidth to the total spectrum bandwidth of the network.

Table 1. Number of rejected requests for network consisting of n=14 nodes.

Load Gbps	m=2			m=4		
	MSP	MSP2	MSP3	MSP	MSP2	MSP3
	Long lived connections, T=10, Length Run=5000					
	2254.6±20.96	2237.7±26.33	2252.7±21.13	832.3±22.75	807.7±19.99	834.0±22.17
	Short lived connections, T=10, Length Run=25000					
1200	128.4±15.00	104.3±13.28	104.0±12.79	0.0±0.0	0.0±0.0	0.0±0.0
1600	809.4±41.96	793.4±40.66	843.1±47.56	1.4±0.89	1.5±0.89	1.5±1.01
2000	2154.0±75.91	2073.7±73.93	2187.1±61.81	11.6±5.39	10.6±4.02	8.4±4.14
2400	3414.6±35.48	3424.3±54.03	3424.3±65.37	50.0±7.48	51.0±7.28	39.0±9.98
2800	4572.4±37.67	4465.7±74.31	4577.6±55.19	161.6±12.06	140.0±15.54	147.4±15.28
3200	6057.2±81.75	5823.4±85.92	6029.5±50.28	632.8±43.54	592.9±27.30	645.5±52.06
3600	6797.5±69.41	6683.9±39.31	6840.6±49.38	1225.2±64.68	1189.8±68.39	1151.9±73.01
4000	7577.4±86.95	7381.5±68.17	7522.5±71.06	1894.9±46.10	1752.5±35.96	1909.5±74.00

Table 2. Capacity blocking probability for network consisting of n=14 nodes.

Load Gbps	m=2			m=4		
	MSP	MSP2	MSP3	MSP	MSP2	MSP3
	Long lived connections, T=10, Length Run=5000					
	0.5057±0.0033	0.4910±0.0044	0.5059±0.0033	0.2014±0.0049	0.1859±0.0035	0.2015±0.0043
	Short lived connections, T=10, Length Run=25000					
1200	0.0082±0.0011	0.0067±0.0010	0.0063±0.0009	0.0000±0.0000	0.0000±0.0000	0.0000±0.0000
1600	0.0519±0.0026	0.0511±0.0025	0.0543±0.0029	0.0000±0.0000	0.0000±0.0000	0.0000±0.0000
2000	0.1359±0.0043	0.1309±0.0042	0.1375±0.0035	0.0007±0.0003	0.0007±0.0003	0.0006±0.0003
2400	0.2101±0.0017	0.2032±0.0031	0.2115±0.0035	0.0031±0.0005	0.0032±0.0004	0.0025±0.0006
2800	0.2756±0.0020	0.2687±0.0038	0.2756±0.0032	0.0101±0.0013	0.0090±0.0009	0.0093±0.0009
3200	0.3553±0.0041	0.3427±0.0040	0.3537±0.0023	0.0407±0.0029	0.0382±0.0017	0.0416±0.0033
3600	0.3934±0.0031	0.3853±0.0013	0.3961±0.0025	0.0785±0.0041	0.0761±0.0042	0.0734±0.0044
4000	0.4328±0.0039	0.4197±0.0032	0.4304±0.0038	0.1194±0.0027	0.1110±0.0022	0.1202±0.0044

Table 3. Spectrum utilization ratio for network consisting of n=14 nodes.

Load Gbps	m=2			m=4		
	MSP	MSP2	MSP3	MSP	MSP2	MSP3
	Long lived connections, T=10, Length Run=5000					
	0.8858±0.0022	0.8892±0.0026	0.8863±0.0018	0.8146±0.0024	0.8152±0.0024	0.8140±0.0023
	Short lived connections, T=10, Length Run=25000					
1200	0.4613±0.0124	0.4672±0.0114	0.5098±0.0129	0.2646±0.0075	0.2639±0.0073	0.2618±0.0082
1600	0.5684±0.0112	0.5601±0.0081	0.5725±0.0053	0.3334±0.0073	0.3337±0.0058	0.3423±0.0077
2000	0.6080±0.0066	0.6063±0.0067	0.6117±0.0034	0.4050±0.0136	0.4002±0.0030	0.4004±0.0103
2400	0.6332±0.0064	0.6287±0.0050	0.6340±0.0075	0.4655±0.0077	0.4719±0.0124	0.4682±0.0082
2800	0.6506±0.0038	0.6544±0.0089	0.6472±0.0088	0.5187±0.0082	0.5335±0.0089	0.5214±0.0092
3200	0.6726±0.0050	0.6760±0.0037	0.6695±0.0030	0.6009±0.0071	0.5908±0.0065	0.6025±0.0065
3600	0.6845±0.0034	0.6894±0.0065	0.6812±0.0042	0.6265±0.0055	0.6225±0.0041	0.6242±0.0047
4000	0.6905±0.0033	0.6956±0.0044	0.6859±0.0030	0.6473±0.0056	0.6436±0.0061	0.6481±0.0045

From Table 1 it can be shown that MSP and MSP3 reject almost the same number of requests in static and dynamic conditions for m =2. The smallest number of requests, in static and dynamic conditions rejects MSP2, which differs from MSP algorithm, the selection scheme of the spectrum feasible segments only from the aggregated spectrum of the path for the incoming request. With the increase of m ($m = 4$), MSP2 algorithm rejects also a smaller number of requests then MSP algorithm in static conditions and in dynamic conditions, especially for higher load of the network (over 1600 Gbps). It should be noticed that under dynamic conditions (m = 4) for several values of load the MSP3 algorithm (with exponential computational complexity) rejects the smallest number of requests, however, this algorithm works without repeatability. Therefore, the obvious conclusion is that the reduction in the number of rejected requests can be obtained not by increasing the computational complexity of the algorithm from the polynomial to exponential (MSP3 or SPV w [1]) but by using the proper selection scheme of the spectrum feasible segments in an aggregated spectrum of the path and an algorithm of polynomial complexity. Table 2 shows the blocking probability obtained after using the examined algorithms. In both static and dynamic conditions the smallest blocking probability is obtained after the using the MSP2 algorithm. Table 3 shows the spectrum utilization ratio after using the examined algorithms. As can be seen, the values of spectrum utilization ratio is similar for all the three examined algorithms in static and dynamic conditions. In turn, table 4 shows the processing time (in ms) after using the examined algorithms. As would be expected, the longest processing times obtained when MSP3 algorithm was used, especially in dynamic conditions of the network. The shortest times were obtained after using the MSP2 algorithm.

Table 4. Processing time for network consisting of n=14 nodes (in ms).

Load Gbps	m=2			m=4		
	MSP	MSP2	MSP3	MSP	MSP2	MSP3
	Long lived connections, T=10, Length Run=5000					
	2186±66.92	2056±49.93	2286±61.69	4831±95.54	4003±69.37	4837±77.02
	Short lived connections, T=10, Length Run=25000					
1200	26807±651.0	23411±161.0	26605±249.8	33523±697.6	28244±310.2	32965±308.6
1600	32113±140.8	27901±357.3	32551±250.4	43854±589.1	33599±413.4	43085±305.7
2000	35947±253.7	30951±292.4	36170±236.0	53192±382.0	39397±589.4	53036±538.2
2400	39173±260.7	33235±258.8	39546±266.6	51674±487.0	46219±501.0	45036±761.1
2800	41446±335.8	34865±147.9	41867±407.2	15791±1042.0	56150±1036.0	30342±545.5
3200	44151±319.8	37426±1048.1	44569±954.0	25700±830.0	14639±1135.0	30390±985.0
3600	45265±387.8	37954±538.4	45719±378.4	23332±864.0	12459±714.0	28363±572.0
4000	46735±400.6	38609±298.2	47352±384.6	26720±1983.0	20433±2106.0	26430±2000.0

A similar relationship, not shown here, were obtained for a network with 16 nodes (the GRID) for which the topological structure is shown on Figure 1.b.

Conclusions

In this work the shortest lightpath selection algorithm that allows realization of the request in flexible transparent optical networks has been proposed. The considered problem covers minimizing the length of the laightpath (transmission distance) being chosen with the constraints imposed on the spectrum continuity and the format of the transmitted signal.

The obtained results showed that the proposed algorithm MSP2, which is based on Dijkstra's algorithm, rejects the smallest number of requests. Furthermore, it was shown, on the example of the MSP3 algorithm, that introduction of the possibility of withdrawing the algorithm from the node with fixed feature generally does not lead to minimization of the number of rejected requests but only to increase the computational complexity of the algorithm from polynomial to exponential. Reducing the number of rejected requests is achieved by using a proper selection scheme of the spectrum feasible segments in an aggregated spectrum of the path, which occupies the minimum segment enabling realization of the incoming request.

REFERENCES

[1] Wan X., Hua N., Zheng X., Dynamic Routing and Spectrum Assignment in Spectrum–Flexible Transparent Optical Networks, Opt. Commun. Netw.,4, (2012), No 8, 603-613

[2] Jnno M., Kozicki B., Takara H., Watanabe A., Sone Y., Tanaka T., Hirano A., Distance adaptive spectrum resource allocation in spectrum-sliced elastic optical path network, IEEE communication Magazine, 48, (2010) No 8, 138-145.

[3] Klinkowski M., Walkowiak K., Routing and Spectrum Assignment in Spectrum Sliced Optical Path Network. IEEE Communications Letters, 15 (2011), No 8, 884-886

[4] Jin Q., Wang L., Wan X., Zheng X., Zhou B., Liu Z., Study of Dynamic Routing and Spectrum Assignment Schems in Bandwidth Flexible Optical Networks, Communicationa and Photonics Conference, (2011), Asia (2007)

[5] Christodoulopoulos K., Tomkos I., Varvarigos A.E., Routing and Spectrum Allocation in OFDM-based Optical Networks with Elastic Bandwidth Allocation. Telecommunications Conference (GLOBECOM 2010), 2010 IEEE 6-10 Dec. 2010.

[6] Christodoulopoulos K., Varvarigos E., Routing and Spectrum Allocation Policies for Time Varying Traffic in Flexible Optical Networks. Optical Network Design and Modeling (ONDM), 16th International Conference, 17-20 April 2012.

[7] Christodoulopoulos K., Tomkos I., Varvarigos E., Elastic Bandwidth Allocation in Flexible OFDM-based Optical Networks. J.Lightwave Technologz, 29 (2011) No 9

[8] Sysło M.M., DeoN., Kowalik J., Discrete Optimization Algorithms: With Pascal Programs, Mineola, N.Y.: Dover Publications, (2006).

***Author**: dr inż. Ireneusz Olszewski, Uniwersytet Technologiczno-Przyrodniczy, Instytut Telekomunikacji i Informatyki, ul. Al. Prof. S. Kaliskiego 7, 85-796 Bydgoszcz, E-mail: Ireneusz.Olszewski@utp.edu.pl*