Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

COMPUTER NETWORK RESEARCH

Exploiting IP-Layer Traffic Prediction Analytics to Allocate Spectrum Resources Using Swarm Intelligence

Constantine A. Kyriakopoulos[1] | Petros Nicopolitidis*[1] | Georgios I. Papadimitriou[1] | Emmanouel Varvarigos[2]

[1]Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

[2]School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece

Correspondence

*Petros Nicopolitidis, Email: petros@csd.auth.gr

Summary

Elastic optical networks emerge as a reliable backbone platform covering the next generation connectivity requirements. It consists of advanced enabling components that provide the ability for extensive configuration leading to performance improvement in many areas of interest. Higher layer analytics like data from IP traffic prediction can assist in the process of allocating resources at the optical layer. This way, light connections are established more efficiently while targeting specific performance goals. For that purpose, an algorithm is designed and evaluated that exploits traffic prediction of data-transfers between nodes of an optical metro or backbone network. Next, it utilizes adaptive functionality based on particle swarm optimization to find paths with available spectrum resources. These resources can facilitate more efficiently the future traffic demand, since traffic prediction data are considered when finding the related paths. The innovative resource allocation method is evaluated using small and very large real topologies. It scales (in execution time and resource usage) according to node-increase, executes in feasible time-frames and reduces transponder utilization resulting to increased energy efficiency.

KEYWORDS:

Particle swarm optimization, linear regression, elastic optical networks, spectrum allocation, grooming, analytics, cross-layer optimization

1 | INTRODUCTION

Optical connectivity enables a reliable data-transfer technology which is designed for covering modern communications between distant end-points like cities or resource-hungry data centers. Most complementary technologies like wireless 5G networking can rely on optical connectivity as an intermediate platform to achieve either transparent or translucent communication. Networking forecasts[1] show that data traffic volume between distant end-points is increasing eventhough less in comparison to the past.

The optical part of this networking landscape improves constantly over time[2], facilitating modern communication needs like high bitrate for video on-demand and access to cloud services. Initially, wavelength division multiplexing (WDM) technology was sufficient enough to offer optical connectivity as a replacement platform to old legacy technologies. High bitrate for transmission, propagation and parallel access to the medium from the end-user application perspective, were key factors for fast adoption. Modern application requirements rely on elaborate resource usage to achieve multiple goals at the same time, like energy efficiency[3], low blocking performance, fast spectrum defragmentation and quality of service (QoS). This

implies that advanced enabling technologies are required, like orthogonal frequency division multiplexing (OFDM)[4,5] and bandwidth-variable transponders (BVTs)[6,7], pushing optical networking to its next phase.

The next generation of backbone connectivity is demonstrated by elastic optical networking (EON)[2]. In this platform, the right amount of resources is provided on-demand to client entities, since BVTs are utilized along with OFDM. The underlying technology relies on the usage of many subcarriers for fulfilling light connections at the optical domain. Their spectra overlap in between, achieving higher overall efficiency and compactness since orthogonal modulation is employed. The enabling technologies are embedded in BVTs for achieving elaborate resource utilization.

Performance improvement is feasible in elastic networking when there is collaboration between the IP layer and lightpath establishment procedures at the lower layer[8]. To achieve this, important data are recorded from the IP layer like the ongoing traffic volume between data-centers. Based on these, analytics are drawn which are useful for improving the efficiency of the lightpath establishment procedures. For example, if large traffic volume is predicted to emerge between two connection endpoints like data-centers, light connections at the lower layer are created in a way by splitting the high traffic volume in different optical routes that share no common intermediate parts. This leads to load balancing that makes the network operate more efficiently.

This procedure can be automated with traffic monitoring at the main backbone nodes, e.g., where data-centers are located. Next, if a threshold is reached in a specific data link, virtual topology reconfiguration is triggered. Finally, the control plane reroutes IP traffic paths in a reactive manner[9].

A similar logic applies to the work of[8]. Virtual topology reconfiguration is possible due to the analysis of big data traffic monitoring. Periodically, traffic prediction is applied and when the virtual topology needs reconfiguration, the optimizer implements a decision-making process which is based on the observe-analyze-act loop[10]. Specifically, a robust and adaptive artificial neural network (ANN) model is used as input to the virtual topology reconfiguration problem and is based on traffic prediction.

These procedures are supported by IETF which standardized the application-based network operation (ABNO) architecture[11]. The ABNO controller is included as a gate in the network architecture relating to actions for coordination. Also, there exists the virtual network topology manager (VNTM), allowing reconfiguration actions on-demand. Next, an element to use for computing paths and a software defined networking (SDN) controller is also described. Finally, a handler for operations, administration and maintenance (OAM) is included as well.

The current work extends the approach of[12]. Traffic analytics are collected by utilizing a linear regression prediction method. Then, a particle swarm optimization (PSO) core utilizes these input data to allocate spectrum resources. In the current form, the main algorithm scales (in execution time and resource usage) to topologies with a very large number of nodes. Evaluation now includes the prediction method which diverges only 2 - 7% in its estimated values from the actual incoming traffic demand.

The contribution of this work relates to the design and evaluation of a spectrum allocation algorithm with the following properties: It operates at two layers simultaneously, i.e., (a) at the IP, where traffic prediction applies based on linear regression, and (b) above the physical layer where the analytics are supplied to the lightpath establishment procedure which is based on swarm intelligence (SI). To the best knowledge of the authors, not many algorithms with this logic exist other than[8] which is resource-heavy, since it utilizes a neural network-based approach to achieve its goals. During the design of the proposed spectrum allocation method, decisions were taken to allow the algorithm to scale upon very large topologies like the 1113-node British Telecom topology. Also, the execution time is adequate to allow either online operation or offline.

The rest of this paper is organized as follows. Metis algorithm along with the problem description are covered in Section 2. The network environment is described in Section 3 and simulation results follow next (Section 4).

2 | METIS ALGORITHM

Metis algorithm (i.e., a mythical titaness -mother of wisdom and deep thought) is described next in detail. Metis (Figure 1) allocates spectrum resources by operating at two layers at the same time. The main functionality is described in Algorithms 2 and 1. When a connection request between a node-pair arrives, a linear regression method predicts the bitrate of the next request to arrive in the future. This is done by examining a window of previous traffic demand values that arrived previously and are logged. Next, the real demand value and the estimated one by the regression method, are transferred to the lower layer as input to the PSO core which receives them for creating a representation of the network topology with updated connection weights. Finally, the PSO core returns the path which serves the performance goals according to the updated topology.

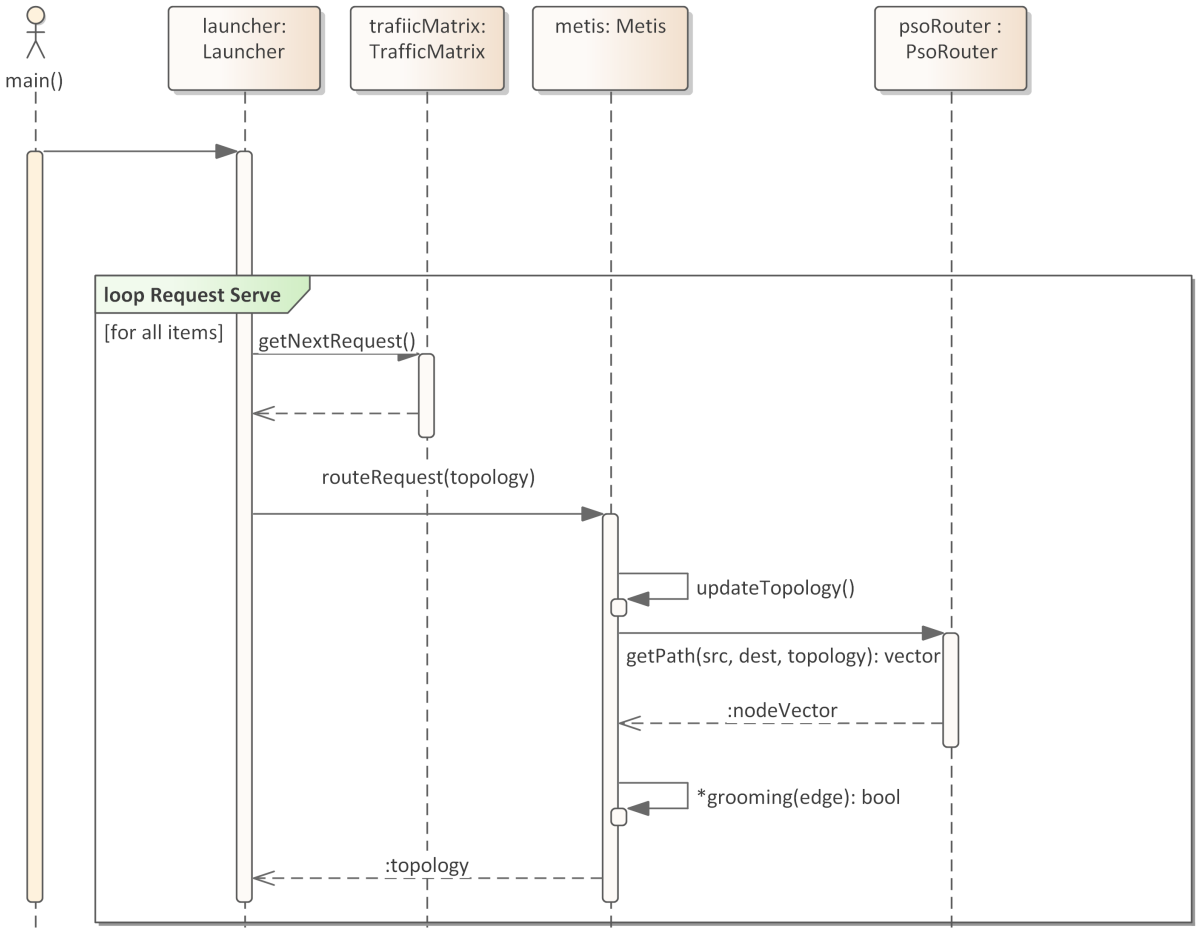

FIGURE 1 Metis sequence UML diagram

2.1 | Problem Description and Network Model

Spectrum resource allocation is the functional goal, when incoming requests arrive in a sequential manner. Problem formulation and the corresponding network model follow next.

- A directed graph describes the physical topology of the elastic network platform, i.e., *G(V, E)*. Therefore, V comprises the set of nodes and E the set of links, nodes connect to.
- A set of frequency slices F are available to initiate light connections between transponders for each link $\in E$. $F = \{f_1, f_2, \cdots, f_n\}$, where n is the maximum number of allowed connections per fiber.
- A set of modulation formats $M = \{m_1, m_2, \cdots, m_n\}$ to use for establishing connections, where n is the maximum number. Specifically, each format is represented by a pair of properties $m = \langle f, r \rangle$, where f is the spectrum of each lightpath and r the corresponding optical reach.
- A set of traffic demands D, i.e., the traffic matrix. Each stored entry is represented by a tuple $d = \langle s, d, b \rangle$, where s is the connection source, d the destination and b the corresponding request's bitrate.

The network model that is followed in this work is described in [13] for the basic RMSA problem.

2.2 | PSO for the Lower Layer

The PSO core routes requests avoiding congested paths by decreasing the evaluation of edges where bandwidth is predicted to increase. A design choice like this leads to a more balanced bandwidth distribution upon the physical topology, when compared to the choice of shortest path for resource allocation. Adaptation to future traffic load is feasible, rendering the algorithm suitable for online operation when runtime conditions may vary. The benefit of PSO over other adaptive solutions is the ability for extensive tuning according to current topology resources. For example, the number of iterations, particles and their neighborhood type, can be tuned specifically for execution with valid results without consuming extraneous computing resources.

PSO emulates the social behavior of, e.g., a flock of birds, as a stochastic optimization method[14]. A particle is an entity representing a solution in the search space. A number of particles cooperate inside an algorithmic flow for positioning close to the best solution. When a number of predefined iterations (i.e., depending on topology resources like the node-number) ends, the particle which provides the best position is the one which solves the problem. A fitness function (in this case Equation 1) evaluates every particle's position at any time. Its velocity (depends on current particle's and neighbors' best positions) also depends on neighbors' positions, and the current and previous best position it had at the moment of evaluation.

Particles communicate between each other in a ring topology, i.e., every particle has two neighbors (Figure 3). When a larger number of particles is used, the probability to find valid paths increases. This leads to more accurate results according to the utilized fitness function. A side effect is the complexity which increases due to the larger number of possible solution evaluations and path constructions inside the particles taking place.

Algorithm 1 Metis Pseudo Code: Lower Layer

```
Update the topology representation with analytics
Setup the PSO core with the updated topology
direct ← false
path = psoRouter(src, dest)
if path.size() > 0 then
    for all edge ∈ pathEdges do
        optical ← false
        if optical = optGrooming(edge) ≠ 0 then
            continue
        else
            direct = directLP(edge)
        end if
    end for
else
    // Use failsafe mode
    Get shortest path from source to destination
    for all edge ∈ pathEdges do
        direct = directLP(edge)
    end for
end if
```

In Algorithm 1, Equation 1 is used for calculating the weights of the updated topology. Since many requests between the same node-pair exist in the traffic matrix, these are used to create analytics for the updated topology. For example, if the fifth request is to be served, the previous four, plus the fifth, comprise the prediction window for estimating the sixth value. If the first four are 10, 20, 30 and 40 Gbps, the current 50 Gbps, the predicted value for the next request is 60 Gbps. The calculated value by Equation 1 will be 55 Gbps. This value, instead of 50, is used to build the updated topology. This way, future traffic demand is served, lowering the blocking probability.

In the activity diagram of Figure 2, the PSO router is responsible for finding the path to fulfill the connection at. If it is valid, optical grooming is attempted on every edge of the path. For the edges where grooming fails, new direct lightpaths are

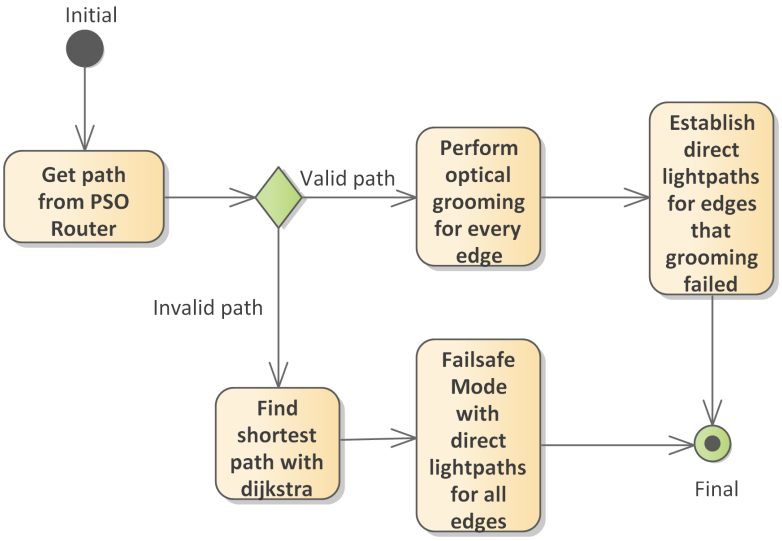

FIGURE 2 Activity diagram for the lower layer

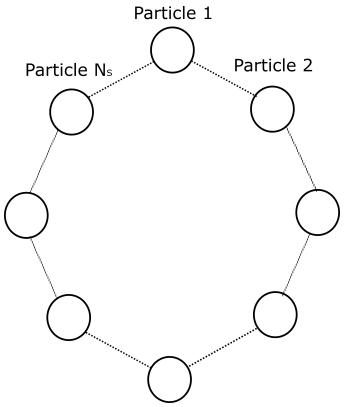

FIGURE 3 Ring particle neighborhood

established with the introduction of new transponders. If the PSO router fails to deliver a valid path, Dijkstra is used for finding the shortest path between request's end-nodes. For every edge of this path, direct lightpaths are established as well.

The PSO core initiates its process of finding the path corresponding to the updated topology. In case the PSO core is set up correctly with enough iterations and particles (depending on topology node-size, in Figure 3), a path is returned suitable for allocating spectrum resources upon. Next, optical grooming is attempted [15] for every physical edge along the path. Thus, available transponder slices at both nodes of the edge initiate and terminate the newly established lightpath [16]. In case no available slice(s) exist at both ends, one (or two) new transponder(s) are used for establishing the new lightpath.

A possible case for the PSO core is not to find a valid path to allocate resources upon. This happens when the number of particles and iterations are not enough in correlation to topology's size. Then, a new lightpath is established that relies on the usage of two new transponders at both ends of every edge. This applies to the shortest path between request's end-nodes.

The pros of using large numbers of particles and iterations lie in the validity of the returned paths. In such cases, these paths are very close to the goal that is fulfilled by the evaluation function. When these parameters exceed certain values according to each topology size, the quality of the results does not improve, but computing resources increase without benefit. So, the right values are part of the overall efficiency.

Algorithm 2 Metis Pseudo Code: Upper Layer

for all $request \in requests$ **do**
// Prediction window with past rate values
$y[...] = oldYs(reqId)$
// Previous time slots corresponding to past rates
$x[...] = oldXs(reqId)$
Get coefficients $a[...]$ corresponding to $y[...]$ and $x[...]$
...by solving Equation 5
Calculate $\hat{rate}_{n+1}$ from Equation 2
Supply to PSO the next estimated and current rate
end for

2.3 | Linear Regression for the Upper Layer

In Algorithm 2, when a new connection request arrives, past bitrate values are collected between the source and destination node. This is the prediction window, and having larger size leads to higher complexity, due to the larger size of matrices in Equation 3. Also, it is not suitable for bursty traffic because a burst can be described with traffic values which are less in number than the prediction window. The goal is to find all $\alpha_0 \dots \alpha_m$ coefficients which are then used in Equation 2 to find $\hat{rate}_{n+1}$. The final step is to supply the real and next estimated values to the PSO core (lower Metis layer) for further processing.

$$Weight = \begin{cases} \frac{rate_n + \hat{rate}_{n+1}}{2}, & \hat{rate}_{n+1} > rate_n \\ rate_n, & otherwise \end{cases} \tag{1}$$

$$\hat{rate}_{n+1} = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots + a_n x^n + \varepsilon \tag{2}$$

$$\begin{bmatrix} rate_1 \\ rate_2 \\ rate_3 \\ \vdots \\ rate_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^m \\ 1 & x_2 & x_2^2 & \cdots & x_2^m \\ 1 & x_3 & x_3^2 & \cdots & x_3^m \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^m \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix} \tag{3}$$

$$\overrightarrow{rate} = X\vec{a} + \vec{\varepsilon} \tag{4}$$

$$\hat{\vec{a}} = \left(X^T X\right)^{-1} X^T \overrightarrow{rate} \tag{5}$$

In Equation 2, $\hat{rate}_{n+1}$ is the estimated edge bandwidth for the next time slot. Previous values from $rate_1$ to $rate_n$ comprise the prediction window, the linear regression method is based on. To calculate $\hat{rate}_{n+1}$, x gets the next slot[17]. The window of previous bitrate values is represented by the vector $rate_1 \dots rate_n$. Equation 3 is the new form of the equation, used to find the window of coefficients $\vec{a}$. Equation 4 is the vector form of the previous representation. Ordinary least mean square estimation is used to calculate the array of coefficients in Equation 5. The outcome is transferred to Equation 2 for calculating the next estimated rate value. Values of ε are possible errors. These are ignored since they tend to small values, not affecting the outcome. Thus, the result is named estimated value.

In the activity diagram of Figure 4, the previous bitrate values that arrived and equal in number to the prediction window, are collected. Each value corresponds to a time slot starting from 1. The array of coefficients a is calculated from Equation 5 for usage in Equation 2. Finally, the latter equation returns the next predicted bitrate value.

3 | NETWORK ENVIRONMENT

Next, the network environment that is used to evaluate Metis' performance is described in detail. It consists of metro and backbone network topologies with various node-sizes, the main optical grooming process, BVTs, modulation formats and the method of generating synthetic traffic.

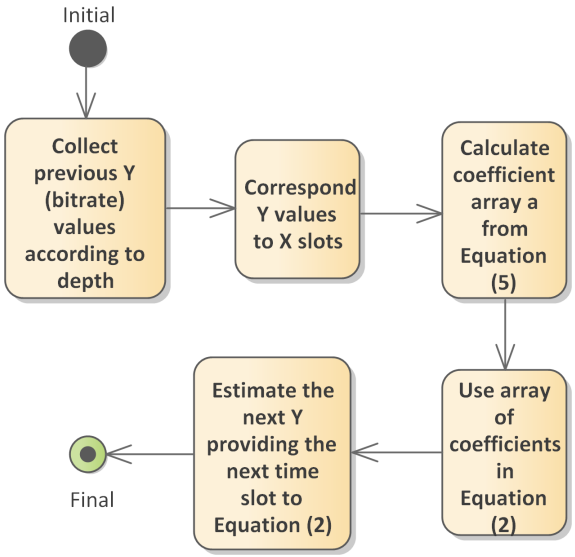

FIGURE 4 Activity diagram for the upper layer

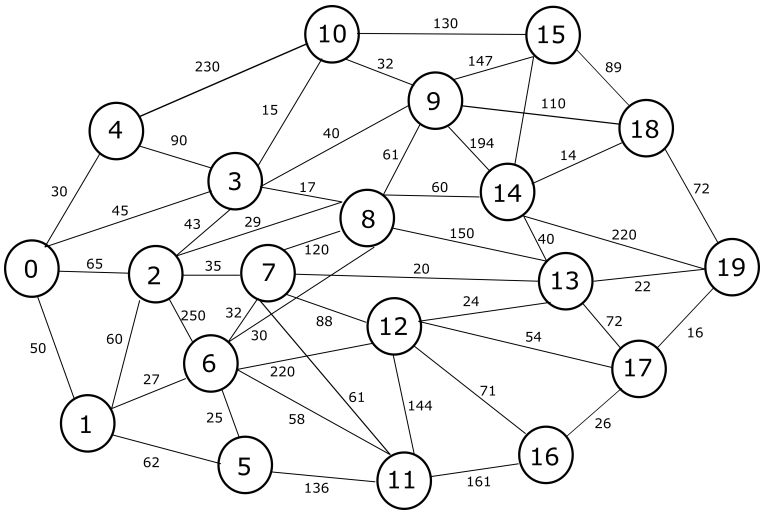

FIGURE 5 20-node topology [19]

3.1 | Topologies

Optical node topologies represent the physical network connections for transferring data between node-pairs, interconnecting at the IP layer. Those used for evaluation purposes are a simple 6-node topology, 12-node Deutsche Telekom (DT) [18], 14-node NSFNet, 20-node (Figure 5) from [19], 29-node Metro (Figure 6) [20], 44-node Telekom Italia (TI) [16] and the 1113-node British Telecom (BT) reference network [21].

Scaling of algorithms while executing on topologies with various sizes, is an important trait for evaluation since it improves their performance and allows stability on environments with higher resource usage. Starting from a simple 6-node topology where a low number of resources is required, the algorithm is evaluated extensively by obtaining the average values of a large number of simulation executions. Next, real topologies like DT and TI provide a test environment for evaluation under real execution conditions. Finally, the very large BT topology provides evaluation under extreme conditions related to heavy resource usage.

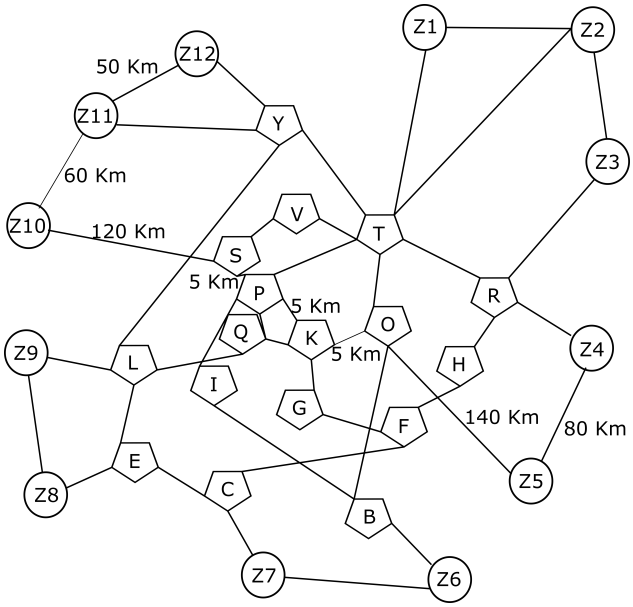

FIGURE 6 Metro topology[20]

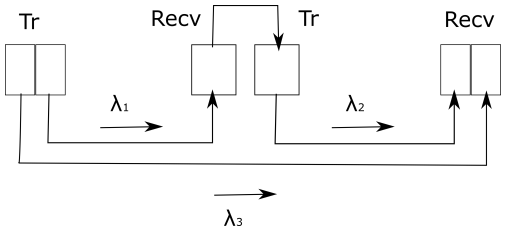

FIGURE 7 Optical grooming

3.2 | Optical Grooming

WDM networks provide the underlying enabling technology for electrical grooming during the spectrum allocation process. This type of grooming is not flexible enough for extensive parameter tuning which affects performance. While performing electrical grooming, Optical-Electrical-Optical (O-E-O) conversions are required at intermediate nodes along a virtual path upon the topology. At the same time, path edges (having transponders at both ends) provide their available bandwidth for grooming by next connection requests. This way, new data transfers at the same virtual path are groomed by using the pre-established lightpaths, as long as the latter provide enough available bandwidth for fulfilling the next request. The benefit from this procedure is the avoidance of new transmitters and receivers at both ends of a lightpath, leading to lower capital and operational expenditures (CapEx and OpEx).

Elastic networks overcome these limitations by providing the ability for optical grooming, enabled by the presence of BVTs. That way, a new lightpath establishment process can benefit from the available subtransponder slices, applying the suitable modulation format according to the physical optical domain distance. All intermediate nodes are bypassed, allowing the optical signal to remain exclusively at the optical domain. So, no new transponders are required along the path, minimizing this way the resource utilization.

Optical grooming is described in Figure 7. Specifically, a new lightpath λ_3 is established when λ_1 and λ_2 already exist in the topology. For optical grooming to succeed, available transponder slices at both ends (left and right nodes) are needed. The optical signal is bypassed at the intermediate node which consists of a transmitter and a receiver.

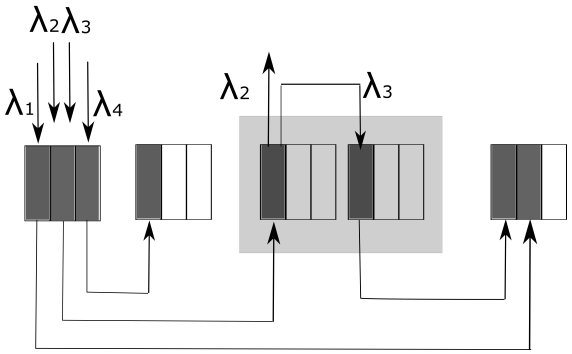

FIGURE 8 BVT example

TABLE 1 Modulation Formats

Modulation Format	*Subcarrier Capacity (Gbps)*	*Distance (km)*
BPSK	12.5	4000
QPSK	25	2000
8QAM	37.5	1000
16QAM	50	500
32QAM	62.5	250
64QAM	75	125

3.3 | Bandwidth-Variable Transponders

Three main types of elastic transponders exist according to their sliceability degree, i.e., non-sliceable (NS-BVT), partially (PS-BVT) and fully sliceable (FS-BVT, used in this research).

The first group consists of transponders that cannot be sliced, since only one available light connection can be created. The ITU-T flex grid is ignored in this case and the spectrum width with the central frequency are utilized by only one lightpath. Since the whole available bitrate is dedicated to this lightpath, e.g., 400 Gbps, this NS-BVT remains lightly loaded in most real operation environments. To compensate, the application of electrical grooming increases resource usage since many data streams are transferred from one lightpath. The role of NS-BVTs is similar to the one of conventional WDM transponders.

The second group contains transponders that only allow a small degree of slicing, i.e., PS-BVTs. This is the reason they are called partially sliceable. Each sub-transponder relies on its own tunable laser, so the cost per transponder provides a mid-term solution to backbone network's resource allocation. What is most important is the introduction of optical grooming capabilities, next to electrical. This allows elaborate resource usage at the algorithmic level.

Finally, fully sliceable transponders are those providing a large sliceability degree, usually more than ten available slices exist. The benefit is that most traffic is off-loaded at the optical layer, leading to less required processing power for electrical grooming. The same transponder is used by many traffic flows from the optical domain perspective, increasing the overall efficiency.

An example of BVTs is illustrated in Figure 8. In this case, lightpaths λ_1 and λ_4 rely on their own slices at the first node. Their destinations are the fourth and second nodes respectively, using only optical grooming. Lightpaths λ_2 and λ_3 share their slice initially, reaching the third and fourth nodes at the end. Processing takes place at first and third nodes due to electrical grooming.

3.4 | Modulation Formats

Elastic networks benefit from the utilization of multiple modulation formats at the same time. This way, resource allocation becomes a process that adapts to the network's operating conditions. Formats are related to the physical reach distance and bitrate to achieve. As a rule, formats supporting higher bitrate can cover shorter distances.

Table 1 provides the available modulation formats Metis uses while establishing lightpaths. Modulation policy is based on these values to occupy the right amount of spectra to create a lightpath between two transponder slices. Subcarrier capacity is limited by the distance factor and is considered for the new lightpath. The process of choosing the right format from the table includes a sort in descending order. The value consisting the ceiling to request's rate is the one the modulation policy chooses. When topologies consist of short distances between nodes, the prevailing format is 64QAM. This leads to the existence of less available slices per transponder since each of these now supports higher bitrate. Network performance improves due to the decreased number of subtransponders consuming energy. Also, the number of line amplifiers decreases as well. Electrical grooming now becomes more prominent in comparison to the optical one since the available bitrate per lightpath is higher.

3.5 | Traffic Generation

A data structure that stores bandwidth requirements between node-pairs is the traffic matrix. For every node-pair there may exist multiple bitrate requests, representing the traffic demand at the IP layer. Each entry is a tuple containing source and destination nodes of each request and the corresponding demand value in Gbps.

Initially, for every source-destination pair a traffic demand value is produced using a uniform distribution. The variable $X \in \{40, 80, 120, 160, 200\}$ in the range $[40, 2X - 40]$, in Gbps. A set of requests belonging to the same node-pair, is produced by utilizing a ZipF distribution. Specifically, the value of X is multiplied to the coefficients that are produced by ZipF. The resulting rate values are allocated to the increasing traffic demand belonging to the same pair.

$$S = \sum_{k=1}^{N} (\frac{1}{k})^{\theta}, \quad Z(i) = \frac{1}{S}(\frac{1}{i})^{\theta} \tag{6}$$

In Equation 6, N is the number of requests per node-pair. The skewness factor is θ which shapes the waveform, and the typical value of 1.2 is used by the current research. The effects to probability waveform different θ values have, is depicted in[22]. Z is the function outcome $\in [0, \dots, 1]$ that is multiplied by the unique value of X allocated to the specific node-pair. The ascending index of bitrate value of the node-pair is i.

4 | SIMULATION RESULTS

Evaluating Metis' performance requires an elastic network environment that enables its functionality. This environment is simulated according to the parameters that are described next in detail.

Lightpaths depend on BVTs which utilize up to 10 connections each. Between them, a guardband of two adjacent frequency slots is present. Each BVT facilitates up to 400 Gbps of data. Modulation formats having different spectrum ranges are chosen on-demand from Table 1 according to the policy described in Section 3.4. A table of data rates is found in[7].

A random function generates the traffic values of $X \in \{40, 80, 120, 160, 200\}$ from the range $[40, 2X - 40]$ in Gbps. Multiple requests per node-pair, where source and destination nodes are uniformly selected, are required according to each execution scenario. A window of five past traffic values assists the linear regression prediction method.

Each execution of the PSO core utilizes a number iterations and population size according to each simulation scenario.

The elastic platform is designed and implemented in Modern C++ with the Clang/LLVM 10 compiler, Boost graph library 1.67 and Armadillo linear algebra library 9. The OS is Debian 10 x64.

In Figure 9, on X-axis is the increasing number of particles, while on Y-axis is the percentage of found valid paths and the percentage of optically groomed lightpaths. As the particle population number increases, both Y variables increase too. More particles allow more iterations to take place, increasing the chances to find valid paths. The variation trend of both curves is similar, since optical grooming applies to paths found by the PSO core. When the PSO core's result is negative, new direct lightpaths are established, introducing new transponders at both ends of each connection request. Execution took place in 44-node Telecom Italia topology due to its large size which may provide failure. The PSO core relies on 20 iterations in this scenario serving 80 Gbps average traffic demand requests. Also, 3 requests are served for each node-pair.

In Figure 10, depicted is the percentage of found valid paths and the percentage of optically groomed lightpaths, according to the increasing average traffic demand. Optical grooming is less feasible when incoming connections require higher bandwidth that the available transponder slices do not support. In this case, new direct lightpaths are established exploiting new resources, previously being idle. At the same time, PSO path-finding percentage is almost stable since the increasing traffic demand does

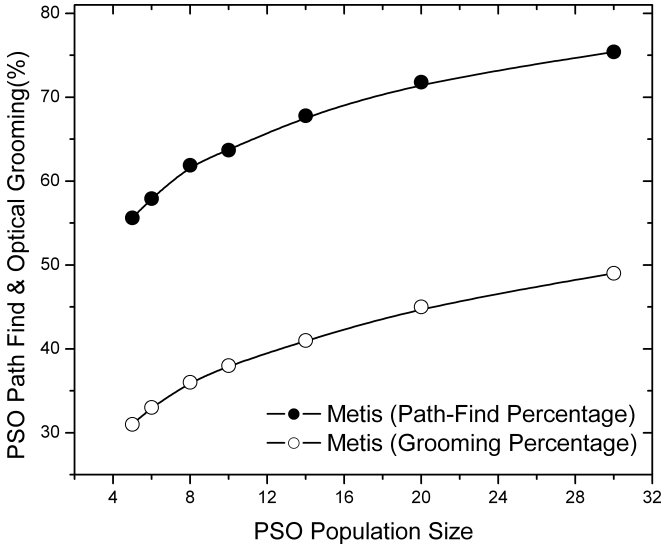

FIGURE 9 Telecom Italia topology path-find percentage

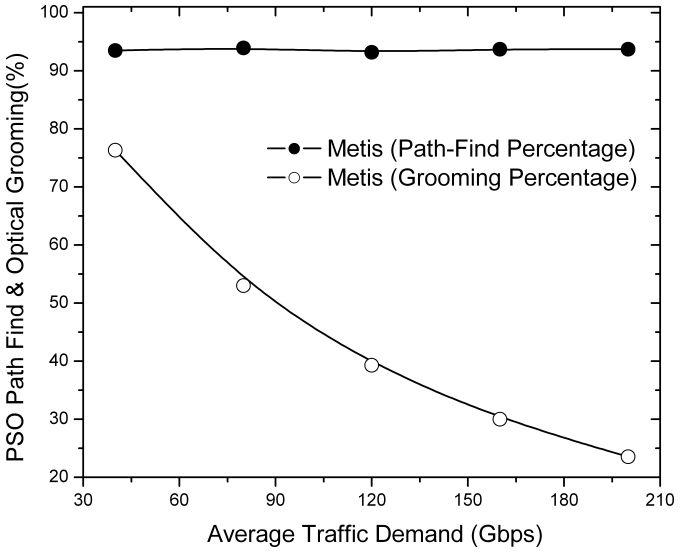

FIGURE 10 Metro topology path-find percentage

not affect the particle population. Execution took place in 29-node Metro topology which is also a large topology. 25 iterations and 10 particles are utilized by the PSO core. Requests per node-pair are 100.

In Figure 11, the ability of Metis to scale is evaluated using different real topologies. Its execution time is depicted according to the node-number of each topology. Each node-pair (i, j) in an N-node topology undergoes 100 served requests from i to j, so the total number of requests is $N \times (N-1) \times 100$. As topology's node-number increases, execution time undergoes polynomial increase. A particle population of 50 entities, along with 200 iterations are used for the PSO core. Average demand is 80 Gbps which is used by the ZipF distribution for curving the demand values for each node-pair.

In Figure 12, the same configuration as in Figure 11 applies to depict the corresponding hop-count numbers. In case the PSO core is replaced by a shortest path algorithm, the average hop-count is becoming larger. This is due to shortest paths being evaluated by their distance and not hop-count. On the other side, evaluation by the predicted bandwidth demand is applied by the PSO core.

In Figure 13, the ability of Metis to predict future traffic demand is evaluated. Requests per node-pair are on horizontal axis. Divergence on the Y-axis is between a predicted traffic value and the actual that arrived next. In case a random selection process

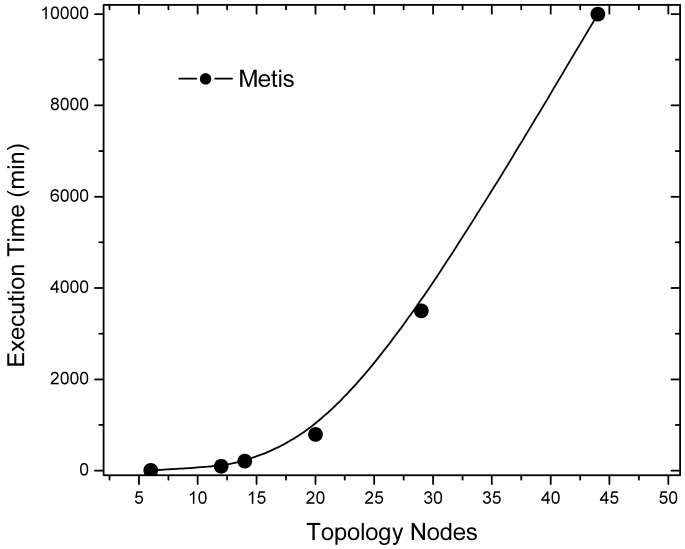

FIGURE 11 Execution time in various topologies

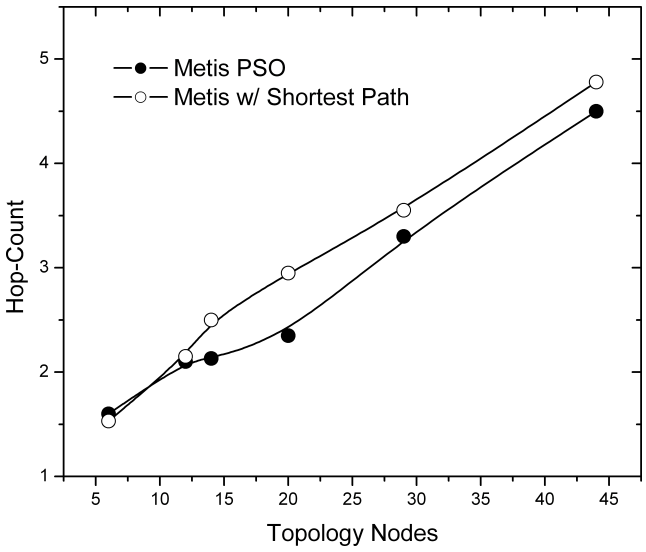

FIGURE 12 Hop-count in various topologies

for incoming traffic demand values is applied (e.g., based on uniform distribution), divergence is higher since the prediction part of Metis undergoes low performance. Predictive methods do not perform well in such cases. In contrast, if the incoming request demand-size is shaped by a ZipF distribution, linear regression returns more accurate results. As the number of requests per node-pair increases, divergence in both cases is lower since the prediction window is full of values. So, linear regression becomes more efficient, predicting with less divergence. The results in this figure are drawn from executing in the 12-node DT topology since low resources ease the evaluation process. 80 Gbps is the average traffic demand, and finally, 50 iterations and 15 particles for the PSO core. The last values are chosen because they lead the optical-find percentage close to 100%.

In Figure 14, the ability of Metis to execute upon a very large topology like the 1113-node BT is evaluated. As the bitrate increases on the horizontal axis, the corresponding hop-count decreases since the transponder limit of 400 Gbps allows less available transponder slices to be used for optical grooming. So, direct lightpaths upon the shortest path become more common. 2000 iterations along with 200 particles configure the PSO core. The corresponding numbers are chosen for every topology since they lead to a path-find percentage being close to 100% (e.g., when the path-find waveform of Figure 9 reaches 100%).

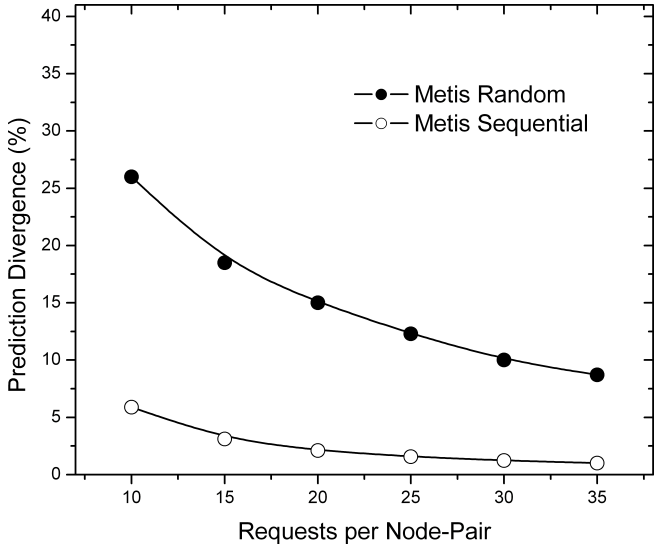

FIGURE 13 Deutsche Telekom prediction evaluation

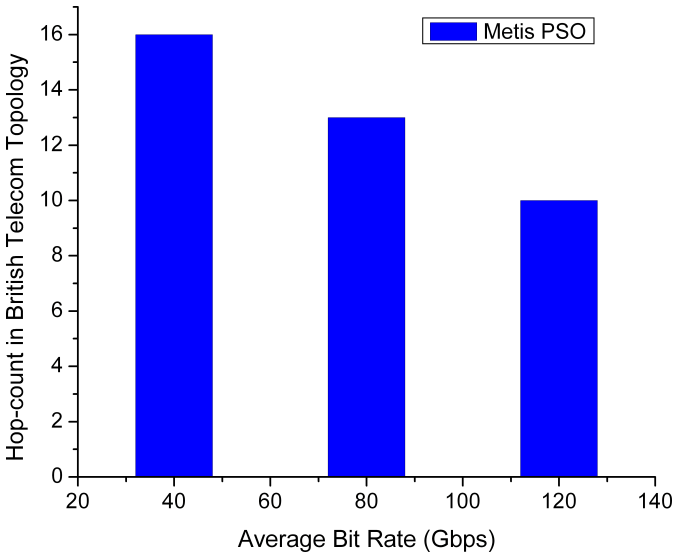

FIGURE 14 BT hop-count

The PSO core in such cases uses the less amount of required resources for returning valid paths, without consuming unneeded extra computing resources.

The main conclusions from results follow next.

- As the particle population size (and the number of iterations[19]) increases, the PSO core return valid paths with higher possibility and the increasing traffic demand is uncorrelated.
- As the average traffic demand increases, less chances exist for optical grooming to succeed.
- Execution time increases according to topology nodes, but their correlation does not impede Metis' practical deployment.
- The average hop-count increases as the size of topology gets larger, but when the average traffic demand reaches transponder's maximum capacity, decreases.
- When the number of requests per node-pair increases, predicted bitrate values are closer to the corresponding real values. Prediction returns more accurate results.

5 | CONCLUSION

Analytics from the IP layer can be exploited while allocating spectrum resources at the physical layer, efficiently. Since data centers operate at a high layer and it is predicted that their traffic volume will keep increasing in the foreseeable future, a new research field is created. Useful traffic analytics are utilized to boost the performance of spectrum allocation. For this reason, adaptivity is required and in this work it is achieved by the usage of swarm intelligence at the lower layer. At the same time, traffic prediction is achieved by utilizing a linear regression method at the higher layer. Performance improves by introducing an algorithm based in this logic. It scales (in execution time and resource usage) according to topology size increase, and efficiently allocates resources with low blocking probability and energy consumption.

ACKNOWLEDGMENTS

This research was co-financed by the European Union and Greek National Funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH-CREATE-INNOVATE (project code: T1EDK-05061).

References

1. Meeker M, Wu L. Internet trends 2018. In: Kleiner Perkins; 2018.
2. Jinno M. Elastic optical networking: Roles and benefits in beyond 100-Gb/s era. *Journal of Lightwave Technology* 2017; 35(5): 1116–1124.
3. Beletsioti GA, Papadimitriou GI, Nicopolitidis P. Energy-aware algorithms for IP over WDM optical networks. *Journal of Lightwave Technology* 2016; 34(11): 2856–2866.
4. Chatterjee BC, Ba S, Oki E. Fragmentation problems and management approaches in elastic optical networks: A survey. *IEEE Communications Surveys & Tutorials* 2017; 20(1): 183–210.
5. Zhang G, De Leenheer M, Morea A, Mukherjee B. A survey on OFDM-based elastic core optical networking. *IEEE Communications Surveys & Tutorials* 2012; 15(1): 65–87.
6. Moreolo MS, Fabrega JM, Nadal L, et al. SDN-enabled sliceable BVT based on multicarrier technology for multiflow rate/distance and grid adaptation. *Journal of Lightwave Technology* 2016; 34(6): 1516–1522.
7. Zhang J, Zhao Y, Yu X, et al. Energy-efficient traffic grooming in sliceable-transponder-equipped IP-over-elastic optical networks [invited]. *Journal of Optical Communications and Networking* 2015; 7(1): A142–A152.
8. Morales F, Ruiz M, Gifre L, Contreras LM, López V, Velasco L. Virtual network topology adaptability based on data analytics for traffic prediction. *IEEE/OSA Journal of Optical Communications and Networking* 2017; 9(1): A35–A45.
9. Aguado A, Davis M, Peng S, et al. Dynamic virtual network reconfiguration over SDN orchestrated multitechnology optical transport domains. *Journal of Lightwave Technology* 2016; 34(8): 1933–1938.
10. Fernández N, Barroso RJD, Siracusa D, et al. Virtual topology reconfiguration in optical networks by means of cognition: Evaluation and experimental validation. *Journal of Optical Communications and Networking* 2015; 7(1): A162–A173.
11. King D, Farrel A. *A PCE-based architecture for application-based network operations*. IETF . 2015.
12. Kyriakopoulos CA, Nicopolitidis P, Papadimitriou GI, Varvarigos E. Adapting Spectrum Resources Using Predicted IP Traffic in Optical Networks. In: SciTePress, INSTICC. ; 2020.
13. Velasco L, Vela AP, Morales F, Ruiz M. Designing, operating, and reoptimizing elastic optical networks. *Journal of Lightwave Technology* 2017; 35(3): 513–526.

14. Zhang Y, Wang S, Ji G. A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering* 2015; 2015.

15. Zhang G, De Leenheer M, Mukherjee B. Optical traffic grooming in OFDM-based elastic optical networks. *Journal of Optical Communications and Networking* 2012; 4(11): B17–B25.

16. Kyriakopoulos CA, Nicopolitidis P, Papadimitriou GI, Varvarigos E. Fast Energy-Efficient Design in Elastic Optical Networks Based on Signal Overlap. *IEEE Access* 2019; 7: 113931–113941.

17. Seber GA, Lee AJ. *Linear regression analysis*. 329. John Wiley & Sons . 2012.

18. Kyriakopoulos CA, Papadimitriou GI, Nicopolitidis P. Towards energy efficiency in virtual topology design of elastic optical networks. *International Journal of Communication Systems* 2018; 31(13): e3727.

19. Mohemmed AW, Sahoo NC, Geok TK. Solving shortest path problem using particle swarm optimization. *Applied Soft Computing* 2008; 8(4): 1643–1653.

20. Antoniades N, Roudas I, Ellinas G, Amin J. Transport metropolitan optical networking: evolving trends in the architecture design and computer modeling. *Journal of lightwave technology* 2004; 22(11): 2653.

21. Klekamp A, Gebhard U. Performance of elastic and mixed-line-rate scenarios for a real IP over DWDM network with more than 1000 nodes. *Journal of Optical Communications and Networking* 2013; 5(10): A28–A36.

22. Kyriakopoulos CA, Nicopolitidis P, Papadimitriou GI. Performance Acceleration in a Push-based Wireless Network Considering Data Item Popularity. In: IEEE. ; 2011: 1–5.