An Online Strategy for Service Degradation with Proportional QoS in Elastic Optical Networks

Alex S. Santos*, Andre K. Horota*, Zhizhen Zhong‡, Juliana de Santi†, Gustavo B. Figueiredo*, Massimo Tornatore¶ and Biswanath Mukherjee§

* Federal University of Bahia, Brazil. Email: santos.alex@ufba.br, {andre.kazuo, gustavo}@dcc.ufba.br

‡ Tsinghua University, Beijing, China. Email: zhongzz14@mails.tsinghua.edu.cn

†Federal University of Technology - Paraná, Brazil. Email: jsanti@utfpr.edu.br

¶Politecnico di Milano, Italy. Email: massimo.tornatore@polimi.it

§University of California at Davis, USA. Email: bmukherjee@ucdavis.edu

***Abstract*—Elastic Optical Networks (EONs) represent a new approach for dealing with the enormous traffic demand in core networks as they can offer bandwidth granularities closer to those requested by the user and hence improve spectral utilization. In current literature there is a lack of dynamic strategies for service degradation which is a possible measure to address problems related to network congestion and consists in reducing the amount of resources provided. Since services of different classes can be requested, we propose in this paper an online strategy for service degradation using proportional Quality of Service (QoS). Our proposed strategy aims at minimizing the number of blocked requests due to lack of resources while provides throughput and delay guarantees for provisioned lightpaths. Thus, in order to quantify the impact of the degradation on the lightpaths we modeled source-destination pairs in an EON as a queuing system working under the Generalized Processor Sharing (GPS) service discipline with admission control of Leaky Bucket policy. The obtained results show that the proposed algorithm can reduce the blocking probability and give network operators more control between different degraded service classes.**

I. Introduction

Despite of the bandwidth flexibility offered by Elastic Optical Networks (EONs), blocking may still occur. Several reasons may originate shortage of resources in an optical network, as failure occurrences (e.g., in case of natural or human-made disasters [1], [2]), or simply the continuous growth of traffic demands, which keeps doubling every two years [3]. Enlarging the network capacity may not always be a feasible option to minimize blocking both from an operational and a financial standpoint. Hence, to maintain revenues and provide clients with a minimum level of service even under network congestion, network operators might decide to degrade the services offered to their clients as an alternative to request blocking. In EONs, such degradation corresponds to deallocating spectrum resources from provisioned lightpaths so that new ones can be accepted.

Some service-degradation proposals have been already published [4]–[6]. But these works employ Integer Linear Programming (ILP)-based solutions that consider static (or quasi-static) scenarios where demands are known in advance. Furthermore, taking into account that different applications might have different requirements, the degradation process should not impact equally all the applications, that is, differentiated treatment must be offered to different classes of services. Some existing works [7], [8] already aim at providing differentiate degradation approaches based on application requirements. However, little attention has been paid to dynamic scenarios where arrival and departure times of lightpath demands are not known in advance. In this scenarios, providing absolute QoS values is hard because lightpath requests arrive one at a time and there is no knowledge about incoming demands. Hence, existing schemes may fail at providing a proper sharing of resources. A possible alternative to absolute QoS is proportional QoS where providers are not required to guarantee strict absolute QoS values. Instead, they only ensure that higher priority classes will receive more resources than lower priority ones according to a proportional relation.

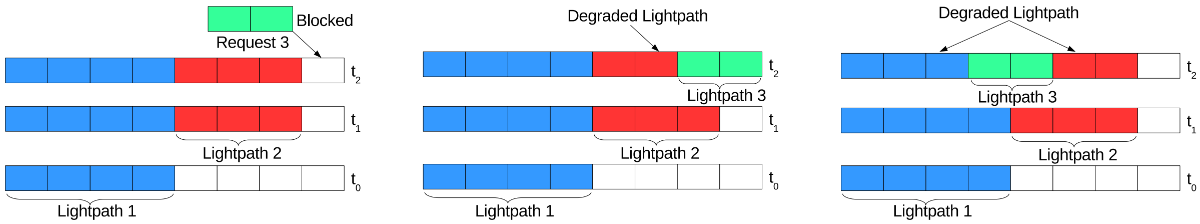
In this paper we introduce a new online algorithm for service degradation that aims at minimizing the blocking probability of incoming requests while ensuring compliance with Service Level Agreements (SLAs). Here, we consider that QoS parameters are translated in a deadline that determines the time for transmission completion. Requests are divided among service classes which will be degraded following a proportional QoS model. To avoid that service degradation does not violate the settled deadline, we model the EON with a queuing system operating with Generalized Processor Sharing (GPS) policy with Leaky Bucket admission control. Results show that the proposed algorithm can reduce blocking probability and gives more control for network operators over service classes.

The rest of the paper is organized as follows: Section II describes service degradation in EONs. In Section III the proposed algorithm is presented. Section IV presents the results obtained. Finally, Section V concludes the work.

II. Related Work

The quality of service offered to each set of applications in a network is described in a SLA that specifies requirements such as priority and delay. While ensuring that these requirements must be met by the network, one of the primary goals of network providers is to maintain their infrastructure profitable and customer loyalty even under shortage of resources. With

978-1-5386-3180-5/18/$31.00 ©2018 IEEE



(a) Lack of frequency slots (b) Single-side degradation (c) Double-side degradation

Fig. 1. Example of service degradation.

the emergence of several traffic-intensive applications, novel measures must be taken to ensure QoS [8], since indefinitely increasing the bandwidth is not an adequate measure due to high costs and different service characteristics. One of such measures is to relax the level of service provided to customers, i.e., a reduction of the amount of resources provided versus those specified in the SLA [6]. This approach is called service degradation. According to [8], degraded provisioning can be implemented either by QoS-assured or QoS-affected policies. In the former, despite the loss of resources, the QoS level is not changed. In the specific case of EONs, the amount of spectral resources removed from each lightpath would be limited by the SLA in such a way that QoS requirements are still met. So, a QoS-assured policy could promote resource compensation either by keeping the transmission rate and adjusting the modulation level or by prolonging the holding time. Both strategies imply releasing spectral resources for incoming requests. In QoS-affected polices, the QoS level can be decreased so the policy can degrade request bandwidth without time or modulation compensation besides not guaranteeing request immediate access. We focus on QoS-assured in this study.

To implement service degradation, it is necessary to choose which lightpaths will be degraded and how a particular lightpath can be degraded while still providing acceptable QoS to the user. The degradation in the optical layer can be done by decreasing the amount of spectral resources occupied by a lightpath. However, to guarantee that the data transmission rate remains unaltered it is necessary to increase its modulation level. Because optical transmission must obey the modulation-distance constraint and high-level modulations tend to have less reach, shorter lightpaths are more likely to be degraded [8]. The degradation can be performed in two ways: single-side degradation and double-side degradation [8]. Two ways of degrading lightpath capacity are illustrated in Fig. 1. Let us consider that request 1 occupies 4 slots and request 2 occupies 3 slots. Given the resource availability in this scenario, request 3 can not be established because it requires two frequency slots (Fig. 1(a)). Fig. 1(b) shows the Single-side degradation, where request 2 gives two slots for request 3. Fig. 1(c) shows the Double-side degradation where both request 1 and request 2 give 1 frequency slot for request 3.

III. The proposed approach

In this section, we present an online strategy to perform provisioning of requests with service degradation in EONs. In the considered dynamic scenario, connection requests arrive one at a time and are represented by the tuple $r_i = (s, d, c, b, w, D_M)$, where s and d are, respectively, source and destination nodes, c is the class type, b is the amount of data to be transmitted, w is the requested bandwidth and D_M the maximum allowed transmission time. Also, all lightpaths in the same source-destination pair are provisioned along the shortest path using the maximum allowed modulation format respecting the transparent reach constraint, which means that all of them use the same modulation format since it is defined based on the path length. In this particular context, there is no gain in reducing the modulation format of lightpaths since the number of slots used would increase. Thus, service degradation is performed by decreasing the number of slots of the lightpath chosen for degradation.

However, at a particular point in time, several lightpaths, possibly more than one from each Class of Service (CoS), might be provisioned along a certain path. So, upon receiving a request and deciding on degradation, network operators must decide which class is going to be penalized, the lightpath to be degraded and the amount of spectral resources that will be taken out of it. In the next subsections, we present our proposed approach to take on these decisions.

A. *The proposed algorithm*

In this section we describe the proposed algorithm, named Minimum Degradation with Proportional QoS (MDP-QoS). The main idea of MDP-QoS is to degrade incoming connections or the already established ones in order to allow the provisioning of incoming requests that otherwise would be blocked due to the shortage of spectral resources. Such service degradation is realized following a proportional relation among the CoS, as described in Section III-B. Moreover, the transmission extension due to the degradation performed must necessarily be within the limits allowed by the SLA of each CoS, as described in Section III-C.

MDP-QoS algorithm is formally described in Algorithm 1. Whenever a new request r_i arrives, the shortest path between the source-destination s-d pair is computed by using Dijkstra's algorithm (Line 2). Then, in Line 3, the maximum allowed modulation format is chosen considering the transparent reach,

and the number of frequency slots (N_l) is computed according to: $N_l = \lceil \frac{w}{M*C} \rceil + N_{GB}$,
where w represents the requested bandwidth, M represents the modulation format, C is the width of a frequency slot and N_{GB} is the guard band.

Algorithm 1: MDP-QoS algorithm

1 Arrival request r_i;
2 Find the shortest-path between source-destination;
3 Define appropriate modulation and calculate N_l;
4 **if** $\exists$ N_l *in path* (s, d) **then**
5   Accept the request r_i;
6 **else**
7   Select the class c_i according to Section III-B;
8   **if** $r_i \in c_i$ **then**
9   Degrade r_i according to its deadline (Section III-C);
10   Number_Of_Request_Degraded_For_c_i++;
11   **if** $\exists$ N_l *in path* (s, d) **then**
12   Accept the request r_i;
13   **else**
14   Select the class c_i according to Section III-B;
15   Go to line 17.
16   **else**
17   Search established requests r_j in path (s, d) belonging to c_i to be degraded;
18   **if** $\exists$ r_j **then**
19   Degrade r_j according to its deadline (Section III-C);
20   Number_Of_Request_Degraded_For_c_i++;
21   Accept the request r_i;
22   **else**
23   The request r_i is blocked;

If there are available frequency slots to accommodate the new request r_i among the s-d path (Line 4), r_i is accepted (Line 5). Otherwise, the algorithm selects the class of service c_i to be considered for degradation taking into account the QoS proportionality (Line 7). The proposed algorithm tries to degrade first the incoming connection r_i (Line 8 - Line 12). Such degradation consists in removing from r_i the maximum number of frequency slots that does not violate its specified deadline (Line 9). An already established connection r_j (Line 17 - Line 21) is degraded only when the incoming request r_i does not belong to the class of service to be degraded c_i (Line 16) or when it is not possible to find a path to provision r_i (Line 13), even performing the degradation according to the specified deadline (Line 9). In case of degradation of both r_i and r_j, a different class c_i is selected for degradation (Line 14) in order to guarantee a proportional degradation among the classes. Furthermore, to be degraded (Line 19), an already established connection must belong to the selected class c_i and be in the path of s-d (Line 17). In order to verify the proportionality of QoS according to Section III-B, in Lines 10 and 20, the number of degraded connections for each class c_i is updated. A connection request is blocked when is not possible to find a path with its minimum allowed bandwidth (Line 23).

B. Proportional QoS Model

A proportional QoS model [9] was adopted to avoid over-penalization of a particular CoS and to determine which light-paths are eligible for degradation. This model is a controllable and scalable solution for quantitative service differentiation. It ensures that higher priority classes will receive better services than lower priority ones according to a proportional relation. Considering q_i a QoS metric of interest and s_i the differentiation factor (i.e. the factor of each class that establishes its proportionality level) for class i, the proportional QoS model forces that:

$$\frac{q_i}{q_j} = \frac{s_i}{s_j}, (i, j = 1...N) \tag{1}$$

Depending on the arrival of demands, there is no guarantee that the proportionality relation among CoS will be continuously sustained. Hence, for practical purposes, in a short time τ the system must guarantee that:

$$\frac{\bar{q}_i(t, t+\tau)}{\bar{q}_j(t, t+\tau)} = \frac{s_i}{s_j}, (i, j = 1...N), \tag{2}$$

where $\bar{q}_i(t, t+\tau)$ is the mean of the QoS metric of interest. From Eq. 1 we have:

$$\frac{a_1 q_1}{a_1 s_1} = ... = \frac{a_n q_n}{a_n s_n} = \frac{a_1 q_1 + a_2 q_2 + ... + a_n q_n}{a_1 s_1 + a_2 s_2 + ... + a_n s_n} \tag{3}$$

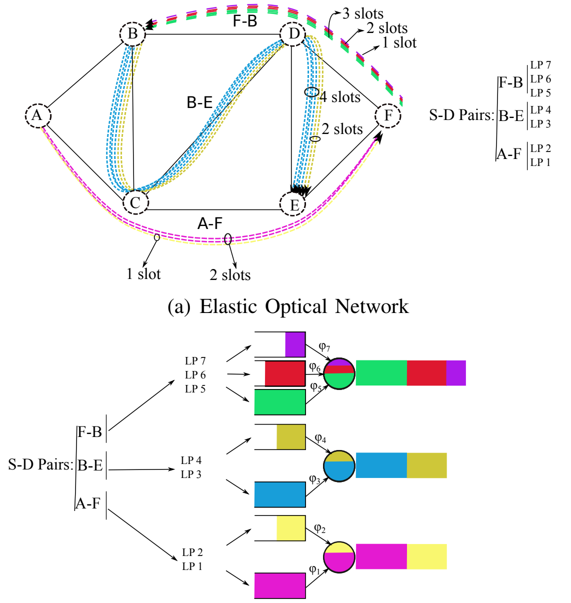
where a_n represents the number of lightpath requests arrived for class n. In other words:

$$q_i^* = s_i \frac{\sum_{k=1}^{N} a_k q_k}{\sum_{k=1}^{N} a_k s_k} = \frac{s_i}{\sum_{k=1}^{N} \frac{a_k}{a_T} s_k} \frac{\sum_{k=1}^{N} a_k q_k}{a_T} = \frac{s_i}{s_{wt}} q_T \tag{4}$$

where q_i^* is the expected value of q_i to assure proportionality, q_T and s_{wt} represent the total number of degraded lightpaths and the sum of weighted differentiation parameters, respectively. They are computed as $q_T = \frac{\sum_{k=1}^{N} a_k q_k}{a_T}$, $s_{wt} = \sum \frac{a_k}{a_T} s_k$. Eq. 4 is used by the algorithm to determine which class will be penalized as it indicates the number of degradations that a class i should experience in relation to the total arrival of lightpath requests q_T, thus preserving the proportionality demanded by the model.

C. Delay control

After deciding to degrade a given CoS, operators must decide how many slots a particular lightpath can lose without violating the transmission deadline. In order to control lightpaths' maximum tolerable delay, we modeled the set of provisioned lightpaths established along the routes used by



(a) Elastic Optical Network

(b) GPS model

Fig. 2. Mapping between the EON and the GPS model.

each source-destination pair as a queuing system operating under the GPS server policy [10]. Such model provides network operators with a set of equations that can be used to extract information about the relation among provisioned and incoming lightpaths. Following the traditional queuing terminology, the queue server is responsible for serving client's flows according to a given service discipline. The GPS is a queuing service discipline in which client's flows are served *bit-by-bit* according to their normalized weights. It offers rate guarantees (Eq. 5) and a fair upper bound for delay (Eq. 7), if arrivals are constrained by the well-known leaky bucket admission control mechanism. Thus, it is possible to quantify the impact of the provisioning of a new request on the transmission times of the lightpaths already provisioned in the network. In a GPS server that operates at a rate R, a rate R_i is guaranteed for each section of traffic i, as follows:

$$R_i = \frac{\phi_i}{\sum_y \phi_y} R, \tag{5}$$

where $\phi_1, \phi_2, ..., \phi_3$ are positive real numbers that represents the weight associated to each flow i. In an EON, the weight that corresponds to the transmission rate of a lightpath l is given by:

$$\phi_l = \frac{N_l T_l}{R}, \tag{6}$$

where N_l represents the number of frequency slots of the lightpath l, T_l represents the transmission rate of one frequency slot according to the modulation format used by lightpath l, and R represents the total used bandwidth along the path between the source-destination pair. This process is illustrated in Fig. 2. In the figure, it is possible to see 3 source-destination pairs (A-F, B-E, F-B). The former uses the path (A-C-E-F) and has 2 provisioned lightpaths (LP1 and LP2), the second uses the path (B-C-D-E) and has 2 provisioned lightpaths (LP3 and LP4) and the latter uses the path (F-D-B) and has 3 provisioned lightpaths (LP5, LP6, LP7). Fig. 2(b) shows those lightpaths represented in the proposed GPS modeling. For each lightpath a relative weight is given that is, according to Eq. 6, related to its number of frequency slots.

$$D \leq \frac{\sigma_i}{\rho_i} < \frac{\sigma_i}{R_i} \tag{7}$$

The GPS gives a fair upper bound for delay if flows requesting queuing services are constrained by a Leaky Bucket Admission Control (LBAC). The admission control is quite simple under this policy. A flow i is admitted if $\sum_i \rho_i < R$ and $\sum_i \sigma_i < B$, where σ_i is the upper bound for the burstiness of the arrival process, ρ_i is the sustainable average rate and B is the bucket size (Fig. 3). Given these conditions, it can be shown that the maximum delay is upper bounded according to Eq. 7 [10]. In an EON, to control the maximum tolerable delay we assume that each lightpath demand is admitted by the LBAC. We also assume that lightpaths always have more capacity than the aggregated traffic coming from upper layers (there is no backlog) and that a lightpath l transmits at the constant peak rate, hence, $\sigma_l = \rho_l = N_l T_l = R_l$. The interpretation of this fact under the LBAC standpoint is that the token arrival rate is equal to the flow's arrival rate. Hence, under the assumption that the traffic is admitted by a LBAC, we have the upper bound for the transmission delay of the lightpath according to Eq. 7. To determine if after losing slots the lightpath l would violate its maximum allowable transmission delay, we change the parameters of the LBAC and compute the new delay according to Eq. 7. For that, we keep the lightpath transmission rate, R_l, and set the token arrival rate to be equal to $N_l' T_l$, where $N_i' < N_i$ is the new amount of slots of the lightpath l. This action is equivalent to compare the original transmission rate (R_l) of lightpath l to the new transmission rate resulting from the loss. This process is illustrated in Fig. 3. Finally, considering D_{new} as the new delay, we compare it to the pre-established value D_M. If $D_{new} > D_M$, the transmission time would exceed the requested transmission deadline should the lightpath l lost slots.

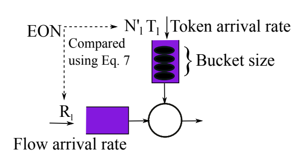


Fig. 3. Leaky Bucket Admission Control.

IV. Numerical Examples

To evaluate the performance of the proposed MDP-QoS algorithm, simulations were carried out. We compared four approaches: Non-degradation (ND), Full Degradation (FD), Partial Degradation (PD) and Minimum Degradation (MDP). The ND approach does not consider degraded provisioning. In the FD approach, incoming requests whose requirements can not be attended are fully degraded, i.e., only one transmission slot and the guard band are left. The scheme PD is similar to the FD, but in this case incoming requests whose requirements cannot be attended are partially degraded (50% or 70% of the requested bandwidth, randomly chosen, is degraded). If necessary, established requests are also partially degraded. Lastly, the MDP degrades only the the minimum required amount of slots. Except ND, all approaches have a QoS-Aware version (FD-QoS, PD-QoS and MDP-QoS) that implement degradation according to the proportional QoS model. Independent replications were performed to generate confidence intervals with 95% confidence level. Ten simulation runs were carried out for each point in the curves, and each run involved 10000 requests randomly generated between all node pairs. Requests arrive following a Poisson process and are uniformly distributed among two classes of service. The bandwidth demands are 5Gbps, 50Gbps and 150 Gbps, and the amount of data to be transferred are 100Gb and 500Gb. The holding time of each request is defined as the ratio between the amount of data to be transferred and its requested bandwidth demand. So, for example, if a service requires 100Gb to be transferred with a bandwidth demand of 50Gbps, the holding time would be of 2 seconds. The deadline for finishing the transmission of each connection requested is generated in the range [25%-100%] of the holding time. Regarding the differentiation factors, $s_1 = 1$ and $s_2 = 2$, i.e., for each lightpath of class 1 degraded, two requests of class 2 should be degraded. The *NSFNET* topology (Fig. 4) with 14 nodes and 21 edges was used in the simulations. Each link in the network supports 300 slots, with 12.5 GHz each. The modulation formats used and their respective transparent reaches (Km) are: BPSK (4000); QPSK (2000); 8-QAM (1000); 16-QAM (500) [11]. For the spectrum assignment, the *First-Fit* policy was used.

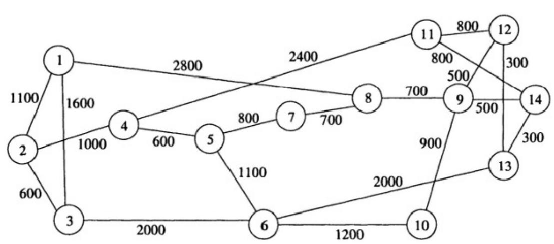


Fig. 4. *NSFNET* topology.

The considered metrics were: blocking probability (i.e., the amount of blocked requests in relation to the total number of requested flows), deadline violation (i.e., the amount of flows that exceeded the specified deadline in relation to the total number of provisioned flows), mean delay (i.e., the additional transmission time due to service degradation) and mean number of degraded connections by CoS. Fig. 5 shows the blocking probability (BP) as a function of network load. The algorithms that degrade connections experience much lower blocking than the one that does not degrade. As the network load increases, such difference becomes even greater due to the inability of Non-Degradation algorithm to adapt the provisioning process to the network conditions. On shortage of resources, approaches without QoS (FD, PD and MDP) degrade the service indiscriminately. Among those, MDP and FD have lower BP. On the other hand, the QoS-Aware approaches (FD-QoS, PD-QoS and MDP-QoS) perform service degradation only if the connection requested belongs to the class that must be degraded in the current moment, i.e., proportional QoS (Section III-B), and only if the amount of degraded bandwidth does not violate the deadline for completing the transmission, i.e., control over delay and degradation (Section III-C). Such types of control are fundamental to ensure the QoS specifications but induce a higher blocking. Among those approaches, MDP-QoS performs better.

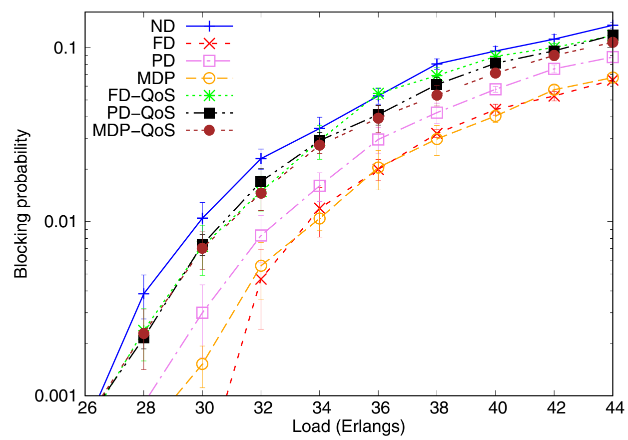


Fig. 5. Blocking probability as a function of network load.

In order to verify the impact of compared algorithms in terms of QoS, we also evaluate the violation of deadline to finish the transmission and the mean enlargement (i.e., the delay) of transmission time as a function of network load (Fig. 6). Since the ND algorithm does not degrade the bandwidth of any connection, it does not violate the deadline or delay the duration of established flows whatsoever. On the other hand, the FD and FD-QoS algorithms can reduce the amount of resources used by the flows regardless their deadline and, as a consequence, have violated the deadline of most established flows. Besides, under loads of 26 Erlangs and 44 Erlangs, FD and FD-QoS have generated mean delay up to $20ms$ and $17.5ms$, respectively, in relation to the specified holding time. PD and PD-QoS algorithms, in turn, by choosing to reduce only part of the amount of resources used by the flows, presented better results for deadline violation, around 80%, and mean delay, with a maximum of $8ms$ approximately, compared to FD algorithm. Conversely, MDP and MDP-QoS algorithms provide control (Section III-B and Section III-C) on how many spectrum slots of a connection can be released in order to avoid the violation of deadline to finish the transmission. Such control of the aggressiveness of degradation avoided the deadline violation for about 98% and

85% of the allocated connections under loads of 28 Erlangs and 44 Erlangs, respectively. Moreover, the generated holding time was delayed in no more than $2ms$ over loads of 44 Erlangs. These results showed that the proposed algorithm slightly impacts the quality of service, which is of paramount importance for both provider's profitability and customer satisfaction.

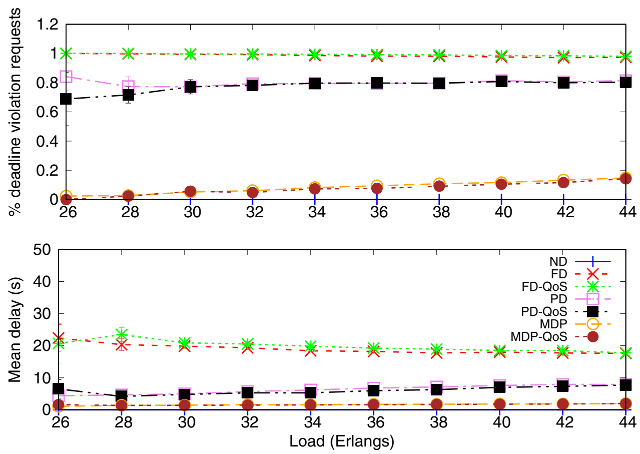


Fig. 6. Deadline violation and mean delay as a function of network load.

Fig. 7 presents the mean number of degraded flows by CoS as a function of network load. The bottom shows the results for the QoS-Aware schemes whereas the top shows for the schemes without QoS. The higher the network load, the higher is the shortage of resources and, consequently, the number of degraded flows also increases. First, it is possible to see that only QoS-Aware approaches can keep proportionality. It is clear that FD, PD and MDP algorithms do not take into account the different levels of degradation allowed by the SLA of Class 1 and Class 2 and, as a consequence, these classes have the same number of degraded connections. This model over-penalizes the connections of one class to detriment of another. By employing the proportional QoS model, QoS-Aware algorithms degrade one connection of Class 1 for each two degraded connections of Class 2, showing that connections are degraded according to the proportionality specified by the Equation 1. Yet, it is noticeable that the control over the allowed degradation provided by MDP-QoS significantly reduces the number of degraded flows. Additionally to the control of deadline violation and delay (Fig. 6), the ability to degrade the service following the proportionality defined is imperative for operators to improve the fidelity of customers as well as to increase revenues.

V. Conclusions

Service degradation is a fundamental alternative to solve problems of network congestion and service provisioning, especially in EONs in which major improvements in terms of transmission capacity and flexibility are expected. This paper investigated the problem of service degradation of incoming and established requests with proportional QoS in EONs. We

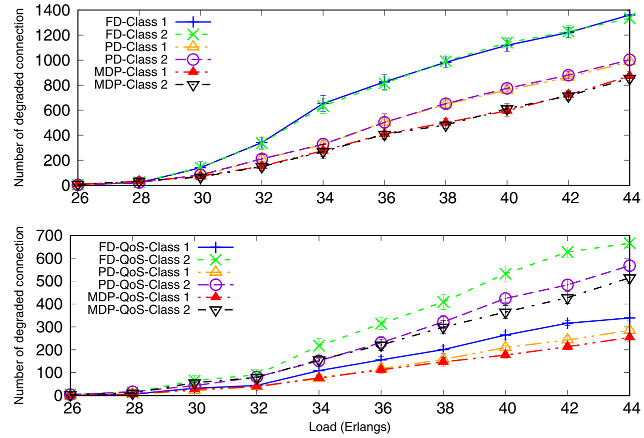


Fig. 7. Mean number of degraded connections by CoS as a function of network load.

proposed an online strategy for service degradation modeled as a Generalized Processor Sharing with Leaky Bucket admission control. Numerical results showed that the MDP-QoS algorithm reduces blocking probability when compared to other QoS-aware approaches, had a significant improvement in terms of quality of service, with 75% less delay when compared to PD-QoS, and also reduces the number of degraded flows when compared to other Qos-aware algorithms.

References

[1] R. E. Krock, "Lack of emergency recovery planning is a disaster waiting to happen," *IEEE Communications Magazine*, vol. 49, no. 1, 2011.

[2] E. Berliner, B. Benmoshe, A. Dvir, and Y. Hadas, "Wireless infrastructure deployment model for sudden-onset disasters through real-time monitoring," in *IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD)*. IEEE, 2016, pp. 194–200.

[3] Cisco, "The Zettabyte Era - Trends and Analysis," 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html

[4] B. Xian, W. Hou, Y. Zong, J. Wu, and L. Guo, "Prediction-based and provident service degradation in optical data center networks," in *Int. Conf. on Optical Comm. and Networks (ICOCN)*. IEEE, 2016, pp. 1–3.

[5] C. S. K. Vadrevu, R. Wang, M. Tornatore, C. U. Martel, and B. Mukherjee, "Degraded Service Provisioning in Mixed-Line-Rate WDM Backbone Networks Using Multipath Routing," *IEEE/ACM Transactions on Networking*, vol. 22, no. 3, pp. 840–849, June 2014.

[6] S. S. Savas, M. F. Habib, M. Tornatore, F. Dikbiyik, and B. Mukherjee, "Network adaptability to disaster disruptions by exploiting degraded-service tolerance," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 58–65, December 2014.

[7] Y. Xiao, C. P. Chen, and B. Wang, "Bandwidth degradation QoS provisioning for adaptive multimedia in wireless/mobile networks," *Computer Communications*, vol. 25, no. 13, pp. 1153–1161, 2002.

[8] Z. Zhong, J. Li, N. Hua, G. B. Figueiredo, Y. Li, X. Zheng, and B. Mukherjee, "On QoS-assured degraded provisioning in service-differentiated multi-layer elastic optical networks," in *IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2016, pp. 1–5.

[9] Y. Chen, C. Qiao, M. Hamdi, and D. H. K. Tsang, "Proportional differentiation: a scalable QoS approach," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 52–58, June 2003.

[10] K.-K. Yap, N. McKeown, and S. Katti, "Multi-server generalized processor sharing," in *Int. Teletraffic Congress (ITC)*, 2012, pp. 29:1–29:8.

[11] N. G. Anoh, M. Babri, A. D. Kora, R. M. Faye, B. Aka, and C. Lishou, "An efficient hybrid protection scheme with shared/dedicated backup paths on elastic optical networks," *Digital Communications and Networks*, vol. 3, no. 1, pp. 11 – 18, 2017.