Restoration Based on Bandwidth Degradation and Service Restoration Delay for Optical Cloud Networks

Felipe Lisboa[1], Keiko V. O. Fonseca[1], Luis C. Vieira[1], Paolo Monti[2], Gustavo B. Figueiredo[3], Juliana de Santi[1]
[1]Federal University of Technology - Paraná (UTFPR) - Curitiba, Brazil
Email: felipe@ecentric.com.br
Email: {keiko, vieira, jsanti}@utfpr.edu.br
[2]KTH Royal Institute of Technology, Sweden
Email: pmonti@kth.se
[3]Federal University of Bahia, Brazil
Email: gustavo@dcc.ufba.br

***Abstract*—The use of an optical fiber infrastructure to accommodate cloud services is gaining a lot of momentum. This is mainly driven by the bandwidth and latency performance that optical transmission can guarantee. On the other hand failures in the optical infrastructure may result in the concurrent loss of a possibly high number of cloud services. For this reason, being able to offer cloud service resiliency at a contained cost is of the utmost importance for operators. This paper proposes a heuristic for the restoration of optical cloud services in the presence of a single fiber link failure. The heuristic leverages on two parameters specified in the service class (i.e., restoration delay and bandwidth degradation) in order to make the best use of the available optical resources during the recovery process. The numerical results presented in the paper show that the proposed restoration algorithm is able to improve cloud service restorability without a negative impact on the cloud service blocking probability.**

I. Introduction

Optical networks are an appealing solution to support a wide range of cloud services in scientific, business, and consumer-based applications (e.g., content delivery services). Large bandwidth availability, low latencies, and reduced energy consumption are some of the important characteristics that make optical networks well-suited for this task [1]. Nonetheless, due to the accidental cuts of fibers, equipment failures, or even malicious attacks, the optical network infrastructure is susceptible to failures. In the presence of a failure, one or more lightpaths are disrupted, potentially affecting several cloud services and consequently causing the loss of a large amount of data. For this reason, network operators must implement recovery schemes to maintain an acceptable level of cloud services survivability while, at the same time, making sure that the resiliency is provided at a contained extra cost (i.e., in terms of how efficiently optical resources are used).

Protection strategies are based on the allocation of redundant optical resources, to be used only in the occurrence of a failure. As a result, these strategies guarantee 100% recovery but have an inherent cost in terms of resource efficiency (i.e., protection resources are most of the time unused) [2]. In order to reduce such cost, operators may use strategies based on the restoration concept. With this approach, no backup optical resources are reserved beforehand. After the occurrence of a failure, the affected lightpaths are re-routed (based only on the available optical resources) in order to restore as many cloud services as possible. As a result restoration strategies are more resource efficient, but cannot guarantee 100% recovery [2].

In the literature, there are a number of studies that try to improve the recovery performance of restoration strategies. Some of them leverage on the concept of cloud service degradation, whenever this possibility is allowed by the specific cloud service class. One possibility is to allow for bandwidth degradation during the restoration process [3]–[6]. Another option is to leverage the maximum allowable restoration delay information (i.e., the time between the occurrence of the failure and the time in which a cloud service is restored) to decide when to restore a cloud service [7]–[12].

Although the aforementioned research directions have been extensively studied, the combined use of both bandwidth degradation and restoration delay has not been investigated. This paper proposes an approach that combines both these concepts to offer a resource-efficient restoration-based recovery strategy for optical cloud services. The joint use of bandwidth-degradation- and restoration-delay-based approaches can be particularly useful since cloud services usually have different requirements [13] which are usually specified in the Service Level Agreements (SLAs) and can be leveraged to decide which strategy to use to recover a cloud service in the presence of a failure. Hard-real-time applications (i.e., surgical procedures) must be immediately restored using an alternative lightpath with enough capacity to recover the entire cloud service. Alternatively, soft-real-time applications (e.g., video streaming) can be restored through an alternative lightpath with a reduced capacity, degrading the cloud service [4] to a minimum acceptable level. On the other hand, non-real-time applications (e.g., grid services for data processing) have some flexibility in both the bandwidth and time domain.

Such flexibility can be used to increase the efficiency of the restoration process (i.e., the number of cloud services that can be recovered) as well as to improve the network resource utilization. For example, the restoration of a cloud service can be postponed based on its recovery time tolerance. By delaying the recovery of non-real-time cloud services, available optical resources can be immediately used for the restoration of critical hard-real-time ones. Moreover, the extra capacity saved (i.e., by allowing the degradation of some cloud services during the recovery process) can be allocated to restore additional cloud services that would have been otherwise dropped. As a result, it is possible to increase the total number of restored cloud services.

The heuristic proposed in the paper builds upon the concepts just explained. After a failure and in the presence of a set of cloud services with different requirements (i.e., non-real-time, hard-real-time and soft-real-time), hard and soft-real-time cloud services can be restored immediately (at full and/or partial bandwidth) not having to compete for resources with cloud services that can be restored at a later point in time. Non-real-time cloud services, in turn, can also benefit from both bandwidth reduction as well as a longer restoration time in order to avoid dropping. To the best of our knowledge, there is no existing restoration strategy based on the combined concept of delayed restoration and bandwidth degradation in Optical Cloud Networks. The simulation results presented in the paper using the NSF topology indicate that the proposed heuristic is able to increase the number of cloud service restored without an impacting on the network blocking probability performance.

This paper is organized as follows. Section II provides an overview of the related work. Section III introduces the proposed algorithm. Section IV presents a numerical evaluation of the proposed algorithm. Finally, in Section V, conclusions are drawn.

II. Related Work

Issues related to the impact of network infrastructure disruptions have motivated the study of strategies for fast and efficient recovery from network failures. Such strategies can be designed considering the different characteristics and requirements of a service. The authors in [13] provide a classification for scientific, business, and consumer applications considering their requirements in cloud and grid systems supported by optical networks. They specify the sensitivity to delay according to that classification.

There is an extensive literature on network recovery based on protection [14]–[16], which guarantee recovery by using redundancy of resources. A less expensive and still efficient approach to provide recovery is to employ service restoration, which tries to recover the services by searching alternative lightpaths reactively to link failures [2]–[5], [17]. The service degradation concept, which guarantees partial amount of the original bandwidth requested by the application, was investigated by Huang et al. [3] for survivable service provisioning schemes and by Savas et al. [4] in admission and recovery of services in order to improve network's adaptability against disasters. Also considering disaster scenarios, the work of [6] proposed a service re-provisioning strategy based on bandwidth degradation and multipath routing aiming to maintain network connectivity and to balance the traffic distribution. The relocation strategy proposed in [17] assumes that a Data Center (DC) node with enough available resources can be used for the service restoration instead of the DC node that was serving the cloud service before the failure. Moreover, Wang et al. [5] presents an integer linear programming (ILP) model based on service degradation and relocation for cloud service restoration. Aiming to minimize both the number of cloud services not restored and the amount of resources used for restoration, a restored cloud service is provisioned using only half of the wavelength capacity. The restoration strategy proposed by the authors in [2] combines the benefits of both cloud service relocation and service differentiation concepts aiming to enhance service restorability making sure that different services receives appropriated priorities.

More recently, shortage of resources in Elastic Optical Networks (EON) has been addressed by the adoption of service degradation strategies. In [18], authors propose a QoS-Assured approach to increase the network acceptance level in an overloaded network. For that the scheme computes the route most suitable for degradation and after that, the level of degradation imposed to the requests is calculated.

The use of deadline information for decision making in systems involving clouds and optical communications networks is an active field of research [6]–[12]. The authors in [7], [9] provided flexible transmission rate and flexible time allocation in Wavelength Division Multiplexing (WDM) networks, considering a set of requests with deadline specifications. Khabbaz et al. [10] proposed an analytical queueing model to flexibly schedule incoming jobs with deadline specifications in cloud data centers. Aiming at simultaneously optimizing data center and network resources, the authors in [12] proposed a time dimension model to schedule advance reservation requests with deadline specifications. The authors in [8] considered that inter-DC connection establishments are delay tolerant and the leftover bandwidth can be reused to complete data transfers.The authors in [16] proposed a RMSA (Routing, Modulation, and Spectrum Assignment) heuristic for survivable transfer on elastic optical inter-DC networks considering a set of requests for bulk data-flow transfer with a time frame in which the transmission must be finished. Considering information about disaster alert and evacuation deadline, the authors in [11] presented a proactive heuristic to evacuate vulnerable and critical content from probable disaster affected DC to safe locations by using the optical path that maximize the amount of data evacuated. They classify the content vulnerability based on the presence of most-updated replicas DC in the disaster zone.

Differently from existing restoration-based recovery strategies, the algorithm proposed in this paper explores the flexibility resulting from the combination of bandwidth degradation and restoration delay in order to increase the chances of

restoring services in optical cloud networks. The service requirements are specified considering different classes of services.

III. Proposed Algorithm

The algorithm proposed in this section, named **R**estoration of **D**ifferentiated Cloud Services based on Bandwidth **D**egradation and Restoration **D**elay (R3D), takes advantage of the fact that the different cloud services requirements can be leveraged upon to allow bandwidth degradation and different levels of restoration delay during the recovery of disrupted services. A delayed restoration occurs only for non-real-time services, while bandwidth degradation can be applied to both non-real-time and for soft-real-time services. It is important to keep in mind that the level of service degradation is always within the limits allowed by the SLA of each service.

In the following sections, we describe the service request model, the classes of service used and we formalize the description of the proposed algorithm.

A. Service Establishment

In the scenario considered in the paper, client nodes require cloud services composed of storage and computation facilities that are provided at geographically distributed Data Center nodes. An optical network (WDM or EON) interconnects client nodes and DC nodes. Moreover, service requests are classified into Class of Services (CoS) according to their tolerance to bandwidth degradation as well as to their tolerance to restoration delay (Section III-B).

Requests for cloud service establishment arrive dynamically. Each request $r(b, h, vm, st, cl, RD)$ specifying the demanded bandwidth (b), the connection holding time (h), the number of virtual machines (vm) and storage units (st), its service class (cl) as well as its level of tolerance to restoration delay (RD). The classes and priorities concepts are used only to restore services disrupted by failures, and they have no influence on how the services are provisioned in the network.

Upon the arrival of a request for service establishment, the closest DC node with enough available virtual machines and storage units and network resources along the route is chosen to support the connection. If no DC node or no lightpath with available resources can be found, the request for cloud service is blocked.

B. Cloud service priority and requirements

In general, after a failure the available optical resource might not be sufficient to recovery all affected services. Therefore, having in place a differentiated restoration policy is essential to improve the number of restored services while guaranteeing that the SLA requirements of those services that are recovered are not violated. Considering the cloud service classification presented in [13], Table I illustrates the requirements and the priorities of the cloud services considered in this work. The table presents three classes of services: hard-real-time, soft-real-time and non-real-time. Upon detection of a failure, the hard-real-time services have the highest priority in the restoration process and need to be recovered immediately (i.e., delay in the restoration is not acceptable) with full bandwidth (i.e., bandwidth degradation is not acceptable).

TABLE I
Cloud service priority and requirements.

Real-time	Priority	Bandwidth degradation	Restoration delay
Hard	High	Not acceptable	Not acceptable
Soft	Medium	Half	Not acceptable
Non	Low	Half	Acceptable

The soft-real-time service class is sensitive to restoration delay while allowing reduction of half bandwidth for restoration. The non-real-time service class has the lowest priority and is recovered only after the restoration of services of all other classes. Moreover, this class allows both bandwidth degradation and restoration delay, being the most flexible class and so providing the greatest opportunities to be explored in our strategy.

C. Service Restoration

The R3D employs an auxiliary graph G to represent the optical resources available after a failure. Services are restored based on their priority, as described in Section III-B. Bandwidth degradation occurs only if there is no enough available bandwidth for a full restoration of soft-real-time and non-real-time services. Moreover, for the services in the non-real-time class restoration is delayed only after an unsuccessful bandwidth degradation attempt.

The R3D restoration algorithm, formally presented in Algorithm 1, is executed whenever a set $S(l)$ of provisioned cloud services are disrupted due to a failure on link l. In Line 1, it is constructed the auxiliary graph G with the available optical resources to be used in the restoration process. In Line 2, the services in S are sorted in descending order of their priority ensuring that hard-real-time services have precedence over soft-real-time and non-real-time services when they compete for the same resources. In the same way, soft-real-time have precedence over non-real-time services.

For each s_i in S (Line 3), the class it belongs to is verified in order to guarantee the appropriate restoration procedure (Line 4, Line 9 and Line 17). If s_i is a hard-real-time service (Line 4), there is no tolerance to bandwidth degradation nor to restoration delay so a lightpath lp with at least b units of available bandwidth must exist (Line 5) to be able to restore s_i by using the bandwidth resource b (Line 6). Otherwise, the requirements for the service s_i can not be met and this hard-real-time service is dropped (Line 8).

If s_i is a soft-real-time service (Line 9) and there exist a lightpath lp with at least b units of bandwidth (Line 10) s_i is restored (Line 11). If this is not the case the algorithm checks the existence of a lightpath lp with at least half of the requested bandwidth (Line 12) to restore s_i with a bandwidth degradation (Line 13). In this case, there is an enlargement of service holding time (Line 14) to compensate for the reduction in the transmission rate. Note that s_i is degraded only if the

available bandwidth is lower than the requested bandwidth ($B < b$). Moreover, s_i can be degraded only once during its provisioning. If there is no path with at least $b/2$ units of bandwidth, restoration becomes unfeasible since is not possible to guarantee the minimum requirement and the soft-real-time service is dropped (Line 16).

If s_i is a non-real-time service (Line 17) there is flexibility in terms of both bandwidth degradation and delay in restoration. In case of availability of resource (Line 18), s_i is restored using the requested bandwidth b (Line 19). If there is only half of requested bandwidth (Line 20), s_i is restored with $b/2$ units of bandwidth (Line 21) and the appropriated extension on its transmission duration is performed (Line 22). If there is a lack of available bandwidth, but there is some tolerance for restoration delay RD (Line 23), the restoration for the cloud service s_i is postponed and a new attempt for restoration is scheduled at time RD (Line 24). Note that a restoration after RD units of time must meet the requested bandwidth b as well as the remaining service holding time. Furthermore, a non-real-time service can be degraded only once during its provisioning. For non-real-time services, there is dropping only in case where there is no available bandwidth and restoration delay has ended (Line 26).

Algorithm 1 **R**estoration of **D**ifferentiated Cloud Services based on Bandwidth **D**egradation and Restoration **D**elay (R3D)

Require: Set $S(l)$ services affected by a failure on link l. Network graph G = (V , E)

Ensure: Each service s_i in S is restored, dropped or has its restoration postponed.

1: Construct an auxiliary graph $G(N, L)$ with the post-failure available optical resource, where N is the set of nodes and L is the set of paths connecting the nodes in N
2: Sort the set S in descending order of priority
3: **for** each service s_i in S **do**
4: **if** $s_i \in$ Hard-Real-Time **then**
5: **if** $\exists\, lp \in G \mid B(lp) \geq b$ **then**
6: Restore service s_i on lightpath lp using b units of bandwidth
7: **else**
8: Drop s_i
9: **else if** $s_i \in$ Soft-Real-Time **then**
10: **if** $\exists\, lp \in G \mid B(lp) \geq b$ **then**
11: Restore service s_i on lightpath lp using b units of bandwidth
12: **else if** $\exists\, lp \in G \mid B(lp) \geq b/2$ **then**
13: Restore service s_i on lightpath lp using $b/2$ units of bandwidth
14: Update the duration of s_i accordingly
15: **else**
16: Drop s_i
17: **else if** $s_i \in$ Non-Real-Time **then**
18: **if** $\exists\, lp \in G \mid B(lp) \geq b$ **then**
19: Restore service s_i on lightpath lp using b units of bandwidth
20: **else if** $\exists\, lp \in G \mid B(lp) \geq b/2$ **then**
21: Restore service s_i on lightpath lp using $b/2$ units of bandwidth
22: Update the duration of s_i accordingly
23: **else if** $RD > 0$ **then**
24: Postpone the service restoration to RD
25: **else**
26: Drop c_i

The construction of the employed auxiliary graph G involves $\mathcal{O}(N^2)$ operations, where N is the number of nodes in the network (Line 1). Considering that the lowest transmission rate is X-OC (Optical Carrier) and wavelength capacity of Y-OC carrier, it can exist at most Y/X disrupted cloud services in the set S, thus there are $\mathcal{O}(Y/X)$ operations in Line 3, where Y/X is a constant. Although the elements of set S are sorted (Line 2), its cost is also constant since there are at most Y/X services in S. To find the path with the requested bandwidth, Dijkstra requires $\mathcal{O}(N^2)$ operations (Line 5, Line 10, Line 12, Line 18 and Line 20). Thus, the complexity of the algorithm R3D is $\mathcal{O}(N^2)$.

IV. Performance Evaluation

The effectiveness of the proposed algorithm is compared with that of a heuristic version of the algorithm in [5], which considers the bandwidth degradation aspect but does not allow delayed restoration. The algorithms use the number of hops to select routes and First-Fit for the wavelength assignment.

The WDMSim [19] simulator was used in the evaluation. Ten simulation runs were carried out for each point in the curves, each run involved 10000 requests for cloud services. Confidence intervals with 95% confidence level were established. The NSF topology (Figure 1), with 14 nodes and 42 unidirectional fiber links was used in the simulation. Each fiber carries 16 wavelengths [5], with bandwidth capacity of an OC-192 carrier (10 Gbps) [5]; each node is a partial grooming node with 32 grooming port pairs (input, output) and no wavelength-conversion capability. It is assumed that nodes 3, 4, 10, and 11 are DC nodes, with 3000 storage and 150 processing units each [5]. The number of storage and processing units required by a cloud service is uniformly distributed with average values of 100 and 5, respectively. Requests for cloud services are uniformly distributed among all pairs of nodes. The holding time and restoration delay follow a negative exponential distribution with mean of 60 time units [5]. Requests arrive according to a Poisson process, and their bandwidth demands are distributed according to the following probability distribution: non-real-time (OC-12:5, OC-24:5 and OC-48:5); soft-real-time (OC-12:3, OC-24:3 and OC-48:3); hard-real-time (OC-6:4 and OC-12:2).

The network load is given in Erlangs defined as

$$A = R \times h \times \left(\frac{B}{\lambda}\right)$$

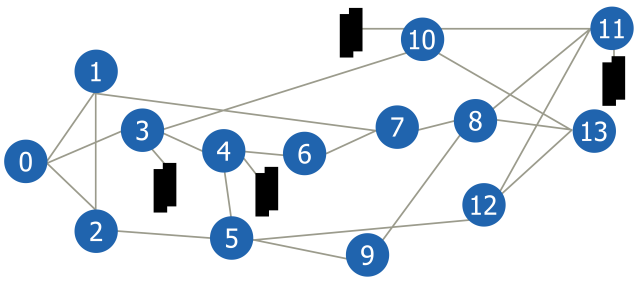

Fig. 1. The NSF topology.

where R is the call arrival rate, h is the call holding time, B is the call bandwidth request normalized to the value of the wavelength capacity λ. Link failures are uniformly distributed over all fiber links. The time between failures is exponentially distributed with a mean value of 1000 time units. The reparation time is also exponentially distributed with a mean value of 10 time units [5]. It is assumed that while one link is under reparation no other links in the network can fail (single link failure assumption).

The metrics collected in the simulations were the mean number of dropped services, the cloud service degradation and the cloud service blocking probability (BP). Dropped services are those affected by failure and not restored. The service degradation is the percentage of restored services with degraded bandwidth or delayed restoration in relation to the total restored services. The cloud service blocking probability is the percentage of blocked services in relation to the total service requested.

Figure 2 shows the mean number of dropped cloud services as a function of the network load. The values for the mean number of dropped services generated by R3D algorithm are considerably lower than those given by the algorithm that does not delay restoration (Degraded algorithm [5]) for all considered loads. Under loads of 30 Erlangs, 700 services

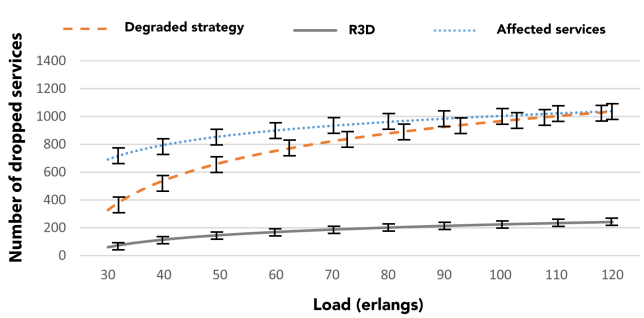

Fig. 2. Mean number of dropped services as a function of network load.

are affected by failure. From this set, the R3D algorithm dropped 55 services, whereas the Degraded strategy dropped 350 services. Under higher loads, the differences between the number of services dropped increase even more. The largest difference between the number of dropped services given by the two algorithms is under loads of 120 Erlangs, when the R3D algorithm dropped 210 services and the Degraded strategy dropped all the 1010 services affected by the failure. The central idea of the Degraded algorithm is to immediately restore the connections for services affected by single failures. Moreover, it can also assign only half of the requested bandwidth to restore soft-real-time and non-real-time services. The bandwidth reduction avoids the drop of these kind of services and maintain minimum acceptable level of quality of service. In addition to the bandwidth degradation for soft-real-time and non-real-time services, the R3D postpone the restoration of connections for non-real-time services affected by failures and, as a consequence, this kind of service has a new chance to be reestablished in near future by using bandwidth released by other services or using the bandwidth from repaired links. Therefore, it is clear that allowing delay in restoration significantly increases the number of restored services.

To evaluate the impact of each kind of degradation on the ability to restore services presented in Figure 2, we also verified the percentage of services restored with degradation of bandwidth or delay as function of network load (Figure 3). Under loads of 30 Erlangs, the percentage of services restored with some degradation were of 2.9% for the R3D algorithm and 0.9% for the Degraded algorithm. Increasing the load to 120 Erlangs, those percentages reach 4.1% and 2.3% for the R3D algorithm and the Degraded algorithm, respectively. Both compared algorithms can restore soft-real-time and non-real-time services by degrading the used bandwidth. Furthermore, the R3D can also rescue non-real-time services affected by link failures by delaying their restoration time, which is indicated by the higher values of service restored with degradation presented in Figure 3.

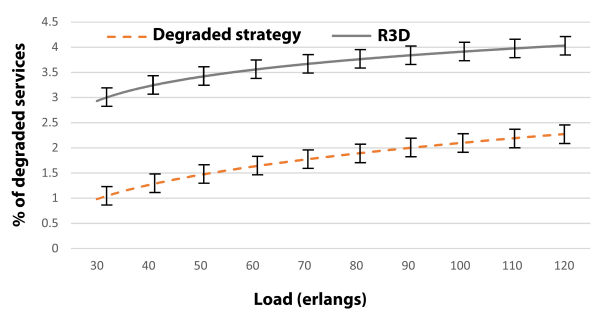

Fig. 3. Cloud service degradation as a function of network load.

Figure 4 presents the cloud service blocking probability (BP) values as a function of the network load. Both the R3D algorithm and Degraded strategy produce acceptable levels of blocking probability. Moreover, our proposed algorithm produced some reduction (about 1.8%) in blocking probability compared to the Degraded strategy.

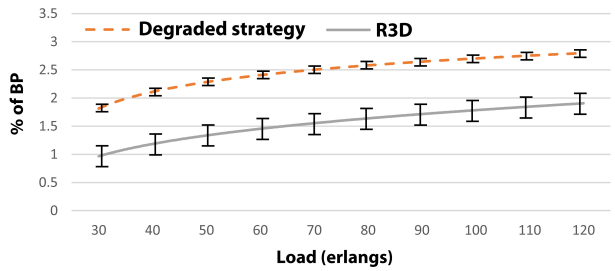

Fig. 4. Cloud service blocking probability as a function of network load.

V. Conclusion

This paper proposes a restoration algorithm (R3D) to recover cloud services disrupted by a single fiber link failure in optical cloud networks. The R3D algorithm leverages bandwidth degradation and service restoration delay, which are determined according to the classes of hard-real-time, soft-real-time and non-real-time services. The algorithm was compared to its counterpart which does not delay restoration. Numerical results show that optical cloud networks can benefit from the proposed algorithm with higher number of restored services. In addition, the results show that the R3D algorithm not only does not bring any increase in blocking probability but also slightly improves it over the compared algorithm.

We are now working on a strategy to degrade existing services with tolerance to restoration delay to release resources in favor of delay-sensitive services disrupted by failures. As future work, we plan to investigate the impact of service establishment driven by network load balancing on restoration ability. Another potential work involves disaster scenarios.

Acknowledgments

This work was partially sponsored by CNPq-Universal under support codes 402805/2016-0 and 420907/2016-5. The authors acknowledge the support of Curitiba Municipality and partners of the project "Smart city concepts in Curitiba – innovation for sustainable mobility and energy efficiency". The latter is funded by VINNOVA (Governmental Agency for Innovation Systems) in Sweden led by KTH Royal Institute of Technology and involving Swedish and Brazilian partners, namely the Federal University of Technology – Paraná (UTFPR), the Volvo Bus Corporation, Combitech, URBS – Urbanization of Curitiba S/A and the city hall of Curitiba. The research has been carried out independently.

References

[1] C. Kachris and I. Tomkos, "A survey on optical interconnects for data centers," *IEEE Communications Surveys Tutorials*, vol. 14, no. 4, pp. 1021–1036, Fourth 2012.

[2] C. N. da Silva, L. Wosinska, S. Spadaro, J. C. W. A. Costa, C. R. L. Frances, and P. Monti, "Restoration in optical cloud networks with relocation and services differentiation," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 8, no. 2, pp. 100–111, Feb 2016.

[3] S. Huang, M. Xia, C. U. Martel, and B. Mukherjee, "A multistate multipath provisioning scheme for differentiated failures in telecom mesh networks," *Journal of Lightwave Technology*, vol. 28, no. 11, pp. 1585–1596, June 2010.

[4] 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.

[5] M. Wang, M. Furdek, P. Monti, and L. Wosinska, "Restoration with service degradation and relocation in optical cloud networks," in *Asia Communications and Photonics Conference 2015*. Optical Society of America, 2015, p. ASu5F.2.

[6] N. H. Bao, M. Tornatore, C. U. Martel, and B. Mukherjee, "Fairness-aware degradation based multipath re-provisioning strategy for post-disaster telecom mesh networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 8, no. 6, pp. 441–450, June 2016.

[7] D. Andrei, M. Tornatore, M. Batayneh, C. U. Martel, and B. Mukherjee, "Provisioning of deadline-driven requests with flexible transmission rates in wdm mesh networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 353–366, Apr. 2010.

[8] Y. Wang, S. Su, S. Jiang, Z. Zhang, and K. Shuang, "Optimal routing and bandwidth allocation for multiple inter-datacenter bulk data transfers," in *2012 IEEE International Conference on Communications (ICC)*, June 2012, pp. 5538–5542.

[9] J. de Santi, N. L. S. da Fonseca, and G. B. Figueiredo, "Algorithm for traffic grooming of batches of deadline-driven requests," in *2013 IEEE International Conference on Communications (ICC)*, June 2013, pp. 2267–2271.

[10] M. Khabbaz and C. Assi, "Impact of job deadlines on the qos performance of cloud data centers," in *IEEE 4th International Conference on Cloud Networking (CloudNet)*, Oct 2015, pp. 32–37.

[11] S. Ferdousi, M. Tornatore, M. F. Habib, and B. Mukherjee, "Rapid data evacuation for large-scale disasters in optical cloud networks [invited]," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 7, no. 12, pp. B163–B172, Dec 2015.

[12] W. Wang, Y. Zhao, J. Zhang, R. He, and H. Chen, "Cross-stratum resource reservation (csrr) algorithm for deadline-driven applications in datacenter networks," *Photonic Network Communications*, vol. 31, no. 1, pp. 162–171, 2016. [Online]. Available: http://dx.doi.org/10.1007/s11107-015-0571-6

[13] C. Develder, M. D. Leenheer, B. Dhoedt, M. Pickavet, D. Colle, F. D. Turck, and P. Demeester, "Optical networks for grid and cloud computing applications," *Proceedings of the IEEE*, vol. 100, no. 5, pp. 1149–1167, 2012.

[14] M. F. Habib, M. Tornatore, F. Dikbiyik, and B. Mukherjee, "Disaster survivability in optical communication networks," *Computer Communications*, vol. 36, no. 6, pp. 630 – 644, 2013, reliable Network-based Services.

[15] C. Natalino, P. Monti, L. França, M. Furdek, L. Wosinska, C. R. Francês, and J. W. Costa, "Dimensioning optical clouds with shared-path shared-computing (spsc) protection," in *2015 IEEE 16th International Conference on High Performance Switching and Routing (HPSR)*, July 2015, pp. 1–6.

[16] N. Wang, J. P. Jue, and R. Zhu, "Survivable bulk data-flow transfer strategies in elastic optical inter-datacenter networks," in *2016 IEEE Global Communications Conference (GLOBECOM)*, Dec 2016, pp. 1–6.

[17] J. Ahmed, P. Monti, L. Wosinska, and S. Spadaro, "Enhancing restoration performance using service relocation in pce-based resilient optical clouds," in *Optical Fiber Communications Conference and Exhibition (OFC), 2014*, March 2014, pp. 1–3.

[18] 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 *2016 IEEE Global Communications Conference (GLOBECOM)*, Dec 2016, pp. 1–5.

[19] A. C. Drummond, "WDMSim - optical WDM networks simulator," 2017. [Online]. Available: http://www.lrc.ic.unicamp.br/wdmsim