

A RELOCATION-BASED HEURISTIC FOR RESTORING OPTICAL CLOUD SERVICES

Carlos Natalino¹, Jawwad Ahmed², Paolo Monti³, Lena Wosinska³, Renato Frances¹

¹ Federal University of Pará, Belem, Para, Brazil, {cns, rfrances}@ufpa.br

² Ericsson Research, Stockholm, Sweden, jawwad.ahmed@ericsson.com

³ KTH Royal Institute of Technology, Stockholm, Sweden, {pmonti, wosinska}@kth.se

ABSTRACT

Optical clouds allow for an integrated management of optical transport and cloud resources (e.g., storage and computing resources running on datacenters). In this paradigm the concept of service relocation (i.e., the ability to re-allocate to a different datacenter node an already provisioned cloud service) offers a new way to restore the network upon failure. This paper presents a heuristic (called H_RELOCATION) based on the service relocation concept to be used for the dynamic restoration of optical cloud services. Upon the occurrence of a network failure, H_RELOCATION solves the routing and resource (i.e., transport plus cloud) allocation problem for each disrupted cloud service allowing, if necessary, to relocate some cloud services to different datacenter nodes. The proposed strategy is benchmarked against two optimal restoration strategies based on Integer Linear Programming (ILP) formulations, the first one without and the second one with the ability to use the service relocation idea. Simulation results show that H_RELOCATION offers performance (i.e., blocking probability, restorability levels and number of required relocations) very close to the optimum, offering on the other hand, a significant reduction in the processing time required for each successfully restored cloud service.

Keywords: *Optical transport networks, optical clouds, service relocation, restoration, anycast, datacenters.*

1. INTRODUCTION

The *optical cloud* is a very popular concept by which storage and computing resources, also referred to as *IT resources*, are spread over different datacenter (DC) locations interconnected via high-speed optical wavelength division multiplexing (WDM) links. In this paradigm cloud services are provisioned in an anycast fashion, where only the source node needs to be specified in the routing and resource assignment phase, while any of the DCs can be used to accommodate a cloud service, as long as its IT resource requirements are met. Anycast provisioning has a number of advantages already recognized by optical cloud providers [1-2]. It allows, for example, for the live

relocation of those cloud services that are already provisioned, i.e., a concept also known as *service relocation*. If the DC location is not essential while provisioning a certain cloud service, it becomes then possible to relocate the job over multiple DC locations. This allows for a more efficient management of both cloud and transport network resources. There are also energy benefits that might derive from relocating, whenever possible, a service in a DC powered by green energy sources [3].

Service relocation has also the potential to bring an extra degree of flexibility to survivability strategies. In fact, by opportunely relocating a cloud service it is possible to use a protection path terminating at a DC that is different from the one used by the primary path. These benefits have been assessed in the literature showing the ability of service relocation to improve resource efficiency when used in conjunction with path protection strategies [2]. Another instance in which service relocation might be beneficial is when it is combined with restoration-based survivability strategies. These strategies are very efficient in the way they use backup resources, i.e., they are dynamically provisioned only upon a failure, but restoration-based approaches suffer from a certain risk that the backup network resources might not be available when needed [4]. Service relocation can potentially alleviate this problem. A study in this direction confirms that relocating a cloud service disrupted by a network failure is beneficial in terms of both restorability and average connection availability [5]. The results in [5] are based on the solution of an integer linear programming (ILP) model, which is fairly complex and does not scale well.

In order to fully assess the benefits of combining the cloud service relocation concept with restoration, in this paper we propose an efficient and scalable heuristic algorithm. A simple but powerful strategy is developed, able to jointly solve the restoration and service relocation problem. The proposed heuristic, called H_RELOCATION, selects for each failed cloud service the best combination of IT and network resources such that the number of restored cloud services is maximized, while the number of cloud service that need relocation is minimized. Simulation results show that the performance of H_RELOCATION in terms of both restorability and average connection availability is very

close to the optimal result obtained by the ILP formulation used as a benchmark [5], while offering a significant gain in terms of processing time.

2. RESTORATION WITH SERVICE RELOCATION HEURISTIC

This section first describes the restoration with service relocation problem and then presents a heuristic (i.e., H_RELOCATION) specifically tailored to solve it.

Let $G(N,E)$ be the graph describing the status of the optical transport network after the occurrence of a failure (i.e., where the failed network element(s) are not included in the graph representation). In this work only a single failure scenario is considered, but the problem and the proposed heuristic can be easily extended to a multi-failure scenario. $G(N,E)$ consists of N nodes and E fiber links. N_{DC} represents the set of datacenters nodes ($N_{DC} \subset N$) each one having DC_k^{st} available storage units, and DC_k^{pu} available processing units, with $DC_k \in N_{DC}$. Let Q be the set of disrupted cloud services that need to be restored after the occurrence of a failure in the network, each one ($Q_i \in Q$) requiring Q_i^{st} storage units and Q_i^{pu} processing units. The source node of Q_i is s_i , while d_i represents the DC node serving Q_i before the failure. The auxiliary variable $curRoute$ is used to store the temporary best route to a cloud service.

The objective of the restoration with service relocation problem is to maximize the number of recovered cloud services $Q_i \in Q$ while minimizing the number of service relocations. A heuristic, which solves this problem, is presented next.

Input:

$G(N, E)$, N_{DC} , $Q_i \in Q$ and network state

Output:

selectedRoute if path is provisioned else NULL

Auxiliary:

$curRoute := NULL$

```

1: if ((selectedRoute := shortestPath( $s_i$ ,  $d_i$ )) != NULL)
    return selectedRoute;
2: for each  $DC_k \in N_{DC} \mid (DC_k^{st} \geq Q_i^{st}) \wedge (DC_k^{pu} \geq Q_i^{pu})$ 
3:    $curRoute := \mathbf{shortestPath}(s_i, DC_k)$ ;
4:   if (hopCount( $curRoute$ ) < hopCount(selectedRoute))
       selectedRoute =  $curRoute$ ;
5: if (selectedRoute = NULL) return NULL;
6: else return selectedRoute;

```

Fig. 1: Pseudo-code for H_RELOCATION heuristic

The H_RELOCATION heuristic is described in Fig. 1. In the figure the *shortestPath* function returns the shortest available (i.e., with available wavelengths) path between two given nodes out of a set of pre-computed k shortest paths between them. If no such path exists or if there are no available wavelength resources on any of the pre-computed paths the *shortestPath* function returns *NULL*. The function

hopCount returns the number of hops of a given route, or *MAX_VALUE* when $curRoute=NULL$.

H_RELOCATION tries to restore each cloud service $Q_i \in Q$ sequentially. For each Q_i the heuristic first checks if there is an available path in $G(N,E)$ from s_i to the DC already in use (i.e., d_i). This is done to reduce the number of unnecessary service relocations. If no such path exists H_RELOCATION tries to find an alternative DC with enough storage (i.e., Q_i^{st}) and computing (i.e., Q_i^{pu}) resources able to accommodate Q_i . If more than one DC with enough resources is reachable, the heuristic chooses the one that is the closest to s_i in terms of hop count. Once the new DC is selected the cloud service is relocated and a lightpath from s_i to the new DC is established. In case neither a suitable DC nor an available path to it can be found, the cloud service is dropped.

3. PERFORMANCE ASSESSMENT

This section describes the scenario used to assess the performance of the H_RELOCATION heuristic, and then discusses a number of simulation results.

3.1 Scenario and assumptions

The NSF network (Fig. 2) is considered as a reference topology [5], where nodes 3, 4, 10 and 11 are assumed to be DC nodes due to their high connectivity. Each DC is equipped with 3000 storage units and 150 processing units. All fiber links in the network are bidirectional, comprising 16 wavelengths each (with the exception of

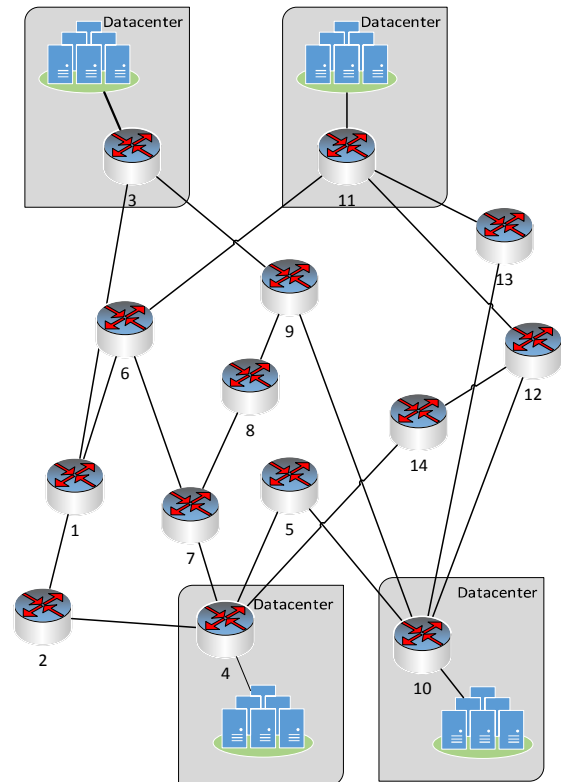


Fig. 2: NSF topology with 4 DCs

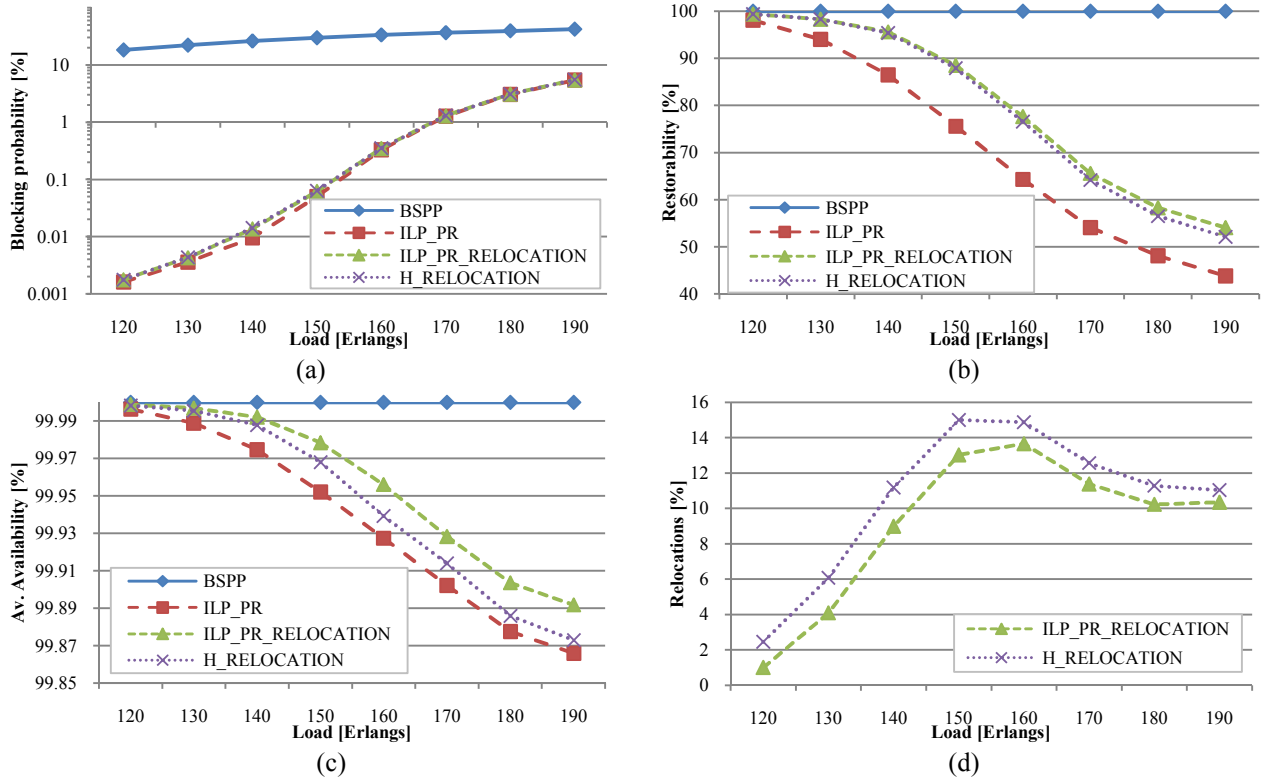


Fig. 3: Simulation results

the link connecting each DC to its network node, which have unlimited capacity). All nodes have full wavelength conversion capability.

Each experiment consists of one million anycast cloud services to be provisioned from a client (i.e., non-DC) node to a DC node (which are uniformly chosen for each cloud service) with enough storage and processing resources, in the interval from 1 to 100 and from 1 to 5, respectively. Each cloud service is also assumed to require the entire capacity of a wavelength channel.

In the normal operating conditions each cloud service is provisioned using the DC_CLOSEST heuristic [6], while upon the occurrence of a failure, H_RELOCATION is applied to restore as many cloud services as possible. Each cloud service holding time is assumed to be exponentially distributed with an average value equal to 60 time-units while service request arrivals at the client nodes follow a Poisson process, with mean time between arrivals defined by the current load. Only fiber link failures are considered in this work. The time between two consecutive failures in the network is exponentially distributed, with a mean value equal to 1000 time units, while the link reparation time, also exponential distributed, has a mean time to repair equal to 10 time units.

All the presented results are the average of 30 different experiments, carried out using a Java-based discrete event-driven simulator [7]. The workstation used is a Red Hat Enterprise Linux with 8 Intel Xeon CPUs (4 cores per CPU) clocked at 2.67 GHz and with 16 GB of RAM memory. The confidence interval of the

blocking probability is 5% or lower, with 95% of confidence level, except for the lower loads.

3.2 Performance evaluation

In the performance evaluation phase three survivability strategies are used for benchmarking purposes. The first one is a protection strategy for anycast cloud services called Backup Server via Physically disjoint Path (BSPP) [8], while the other two are ILP models representing: (i) an optimal restoration strategy without relocation capabilities (i.e., ILP_PR) [4], and (ii) an optimal restoration strategy with relocation capabilities and minimum number of cloud service relocations (i.e., ILP_PR_RELOCATION) [5].

Figure 3 shows some performance results of the H_RELOCATION strategy as a function of the network load. The first thing that can be noticed is the inherent benefit, in terms of blocking probability (Fig. 3a), when using restoration-based strategies. More than one order of magnitude can be gained at low to medium loads when no dedicated protection resources are assigned. In addition it can also be noticed that using cloud service relocation (and consequently potentially longer restoration paths) has a little impact on the blocking probability. Fig. 3b shows the ratio between the number of successfully recovered cloud services over the total number of recovery attempts (i.e., restorability) as a function of load. As expected, BSPP guarantees 100% recovery against any single failure. However, as shown in Fig.3a, this comes at the expense of very high blocking probability, even in relatively low load conditions. On the other hand, the relocation

opportunity is quite efficient in increasing the chances of a cloud service being recovered. In this respect H_RELOCATION presents restorability values that are very close to the optimal (i.e., ILP_RELOCATION). Similar conclusions can also be drawn while examining the average connection availability (i.e., the ratio between the time a cloud service is operative over the entire cloud service holding time) value shown in Fig. 3c. In this case H_RELOCATION is slightly worse than ILP_RELOCATION, which is due to the sequential nature of the heuristic that cannot concurrently optimize the restoration attempt of batch of disrupted cloud services. Finally, Fig 3.d shows efficiency of the proposed H_RELOCATION heuristic in containing the number of required relocations (i.e., ratio between the number of restored cloud service that required relocation over the total restoration attempts) when compared to their minimal value (i.e., ILP_PR_RELOCATION). The graph also shows that when the blocking probability is low, the number of relocations grows with the load, but after the system saturates, the number of relocations stabilizes, showing that the DCs and network resources are already highly utilized and it is difficult to accommodate services to be relocated from the affected DC to another DC, which reduces the number of possible relocations when the load is further increasing.

Table 1 presents a comparison between the proposed heuristic and the ILP based optimal strategy in terms of average processing time for each disrupted cloud service. The analysis is done for three load levels, i.e., low, medium and high. It is possible to see that for the proposed heuristic the processing time, for all load levels, is always below one millisecond, while the optimal strategy requires much higher processing times. This is an important aspect to consider especially in those scenarios where each failure is expected to affect a high number of cloud services (e.g., high number of wavelength channels).

Load [Erlangs]	Processing time [ms]	
	H_RELOCATION	ILP_RELOCATION
120	0.191	25.133
160	0.155	30.876
190	0.262	25.78

Table 1: Average processing time for each disrupted cloud service restoration attempt

The sequential heuristic also makes possible to start the reestablishment of computed restorations and relocations while calculate the remaining disrupted connections, what is not possible on the ILP once it optimizes the entire set of disrupted connections, allowing the reestablishment of connections only in the end of the computation.

4. FINAL REMARKS

This paper presents a sequential heuristic (i.e., H_RELOCATION) aimed at restoring a set of failed

cloud services using service relocation in a dynamic optical cloud network scenario. When benchmarked against optimal solutions provided by two ILP formulations, H_RELOCATION shows restorability and average connection availability values that are very close to the optimum, with only a slightly increase in the number of services that need to be relocated to a different datacenter. In addition the processing time of H_RELOCATION is more than two orders of magnitude shorter than the processing time of the optimal strategies.

5. ACKNOWLEDGMENTS

This work has been supported by FAPESPA Grant no. 005/2011, by CNPq-CISB-SAAB Grant no. 55/2012 and by ICT DISCUS (FP7 program, grant agreement n° 318137).

6. REFERENCES

- [1] C. Devellder et al., "Optical Networks for Grid and Cloud Computing Applications," *Proceedings of the IEEE*, vol. 100, no. 5, pp. 1149,1167, May 2012.
- [2] C. Devellder, J. Buysse and B. Jaumard, "Joint Dimensioning of Server and Network Infrastructure for Resilient Optical Grid/Clouds," *IEEE/ACM Trans. On Networking*, Vol. PP, no. 99, 2013.
- [3] U. Mandal, M. Habib, S. Zhang, B. Mukherjee and M. Tornatore, "Greening the Cloud Using Renewable-Energy-Aware Service Migration," *IEEE Network*, Vol. 27, no. 6, pp. 36-43, 2013.
- [4] J. Ahmed, C. Cavdar, P. Monti and L. Wosinska, "Hybrid Survivability Schemes Achieving High Connection Availability with Reduced Amount of Backup Resources," *IEEE/OSA Journal of Optical Communication and Networking*, Vol. 5, no. 10, pp. 152-161, Oct. 2013.
- [5] J. Ahmed, P. Monti, L. Wosinska and S. Spadaro, "Enhancing Restoration Performance Using Service Relocation in PCE-based Resilient Optical Clouds," *Proc. Of Optical Fiber Communication Conference*, 2014.
- [6] G. Landi et al., "A Network Control Plane Architecture for On-demand Co-provisioning of Optical Network and IT Services," *Proc. Of Future Network & Mobile Summit*, pp. 1-8, July 2012.
- [7] J. Ahmed and S. Nawaz, "POSE: A new discrete event optical simulator for the Java platform," *Proc. of NCET*, Pakistan, Dec. 2004.
- [8] Q. She, X. Huang, Q. Zhang, Y. Zhu and J. Jue, "Survivable Traffic Grooming for Anycast in WDM Mesh Networks," *Proc. Of IEEE Globecom*, pp. 2253-2257, Nov. 2007.