Routing and Scheduled Spectrum Allocation for Transfer-Based Datacenter Connections

A. Asensio*, M. Ruiz, and L. Velasco
Optical Communications Group (GCO). Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
**e-mail: aasensio@ac.upc.edu*

ABSTRACT

Creating datacenter (DC) federations allows DC operators to reduce costs and increase their revenues from using under-utilized resources as well as to expand their geographical coverage aiming to improve users' Quality of Experience. Since huge capacity is needed to interconnect the DCs in a federation, elastic optical networks can be used. Taking advantage of an intelligent control plane, connectivity can be requested on demand. To that end, the Application-Based Network Operations (ABNO) architecture, standardized by the IETF, can be used. Notwithstanding, connection requests can be blocked in the case that no optical resources are available at the time they are needed. In view of that, an Application Service Orchestrator (ASO) on top of the ABNO architecture can be used to coordinate resource utilization among services. Under that approach, DC resource managers request data transferences using an application-oriented semantic, including data volume and completion time. Those requests are transformed into connection requests by the ASO, which solves the Routing and Scheduled Spectrum Allocation (RSSA) problem that might involve performing elastic operations over on-going transferences while ensuring their committed completion time. In this paper, we detail an algorithm for solving the RSSA problem under scenarios where spectral resources are highly utilized reducing thus spectrum fragmentation.

Keywords: transfer-based connections, datacenter interconnection, flexgrid optical networks.

1. INTRODUCTION

Works that can be found in the literature (see e.g. [1], [2]) show that datacenter (DC) federations allow DC operators to reduce costs and increase their revenues from using under-utilized resources as well as to expand their geographical coverage aiming to improve users' Quality of Experience (QoE). A technique for reducing costs and improving QoE is virtual machine (VM) migration between DCs within the federation. To that end, scheduling algorithms inside datacenter resource managers can take decisions related to VM migration and database synchronization based on some cost function, such as minimizing brown energy consumption, while taking advantage of solar green energy availability in remote DCs [1].

As a result of the scheduling algorithms, different connections to remote DCs need to be established and huge volume of data may need to be transferred, e.g. several TB, during given periods of time. From the network operator perspective, the advent of elastic optical networks and intelligent control planes to automate optical connection provisioning, can facilitate network elastic operations according to DC operator's on demand requests. Among intelligent control plane technologies, the IETF has standardized a centralized architecture named Application-based Network Operations (ABNO) [3] able to implement in-operation planning [4].

Taking advantage of elastic operations and ABNO in elastic optical networks, authors in [5] proposed to use a flexgrid-based optical network, where cloud resource managers can request connections of the desired bitrate, and tear them down as soon as they are not really needed. Owing to the fact that network resources might not be available at requesting time, elastic connections were proposed in [5] as a way to reduce the time needed to transfer the volume of data. In [6], cloud resource managers request data transferences to be completed within a maximum completion time (e.g. the period of scheduling algorithms). An Application Service Orchestrator (ASO) on top of ABNO transforms *transfer-based* requests into network connection requests. Figure 1 illustrates the control architecture to manage transfer-based requests.

In this paper we extend our work in [8] where the routing and scheduled spectrum allocation (RSSA) problem and an algorithm for solving it were presented aiming to manage efficiently transfer-based connections for inter-datacenter connectivity. Elastic operations for both increasing and decreasing the allocated spectral resources to a given connection were considered. Moreover, specific elements in ASO to solve the RSSA problem were described; i.e. a specific service database containing information about the services deployed and a scheduled traffic engineering database (sTED) to manage scheduling schemes. An algorithm for solving the RSSA problem for transfer-based inter-DC connections is described for the case where no continuous frequency slots can be found in a given route because of the high spectral resources utilization.

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement n° 317999 IDEALIST and from the Spanish MINECO SYNERGY project (TEC2014-59995-R).

978-1-4673-7879-6/15/$31.00 ©2015 IEEE

2. ROUTING AND SCHEDULED SPECTRUM ALLOCATION

For illustrative purposes and describing the routing and scheduled spectrum allocation problem, let us assume a scenario in where a resource manager in a DC sends a transfer-based request to the ASO, specifying the target DC, the amount of data to be transferred and a maximum completion time to perform the data transference. In the case that no resources are available at requesting time (t^0) to guarantee the required completion time and aiming to make enough room for the incoming transfer, the ASO might try to reduce the resources allocated to other transfers currently in progress while guaranteeing their committed completion time.

Figure 2 shows examples of the described operation based on two different scheduling schemes. The bitrate of the optical connection supporting on-going transfer $r1$ in Figure 2a can be squeezed thus increasing its scheduled completion time t_{r1}, but without violating the committed completion time, c_{r1}. Released resources can be used to set-up an optical connection to support the requested transfer $r2$.

Another example is depicted in Figure 2b, where to ensure the committed completion time for transfer $r1$, some resources need to be returned to that transfer as soon as transfer $r2$ is completed. In view of the example, it is obvious that spectral resources need to be scheduled so as to be allocated to the same transfer in the future. In this case, it is clear that elastic operations include increasing and decreasing connections' bitrate and that after a connection has been torn down, released resources need to be allocated to those connections supporting on-going transferences, so as to satisfy the planned scheduling.

However, some resources might remain unallocated even after scheduled allocations are done. Aiming at maximizing resource utilization, connections supporting on-going transferences are offered the opportunity to increase allocated resources reducing thus its scheduled completion time. Note that the benefit from the latter is not only completion time reduction, but also to accelerate that those connections can be torn down sooner. That effect might be amplified by assigning released resources to those connections whose scheduled completion time is closer to the current time.

Because of the resources assignment policy described in the previous paragraphs, the spectral occupancy in fiber links in a given route may lead to the situation where no continuous frequency slots can be found and therefore a new transference request would be blocked. Figure 3a illustrates an example where no continuous slot can be found due to the spectral resources occupancy. Table 1 shows the algorithm proposed for solving the RSSA problem in this scenario and given a route, p; a border spectrum slice b; the minimum admissible bitrate for the new transference, br; and the previously introduced *sTED*.

Firstly, the required slot width, sw, in the route p is computed (line 1). Then, connections having as border spectrum slice b are selected, re-scheduling for the on-going transferences supported by those connections is evaluated and a slot is generated by reducing the spectrum allocation of those connections (considering the committed completion time for the corresponding transference) (lines 2-3). The resulting available slot, or slots, are found and sorted by its width in descendent order (lines 4-5). For each of them, adjacent connections supporting on-going transferences are considered as candidate to squeeze its current bitrate provided that the committed completion time is satisfied (lines 6-12). Specifically, the *findSlot* function tries to increase the slot width in an attempt to make room for the incoming transference. The needed re-scheduling are afterwards stored in the *sTED* database and connection update requests are sent to the ABNO controller in charge of the optical networks. To compute the maximum bitrate squeezing that can be applied, scheduling schemes introduced above in this section and depicted in Figure 2 are applied. Finally, if a feasible slot is not found, the algorithm returns *NO_FEASIBLE_SOLUTION.*

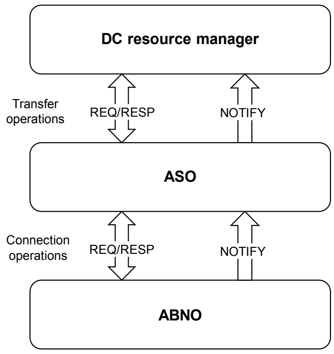

Figure 1. Control architecture scheme.

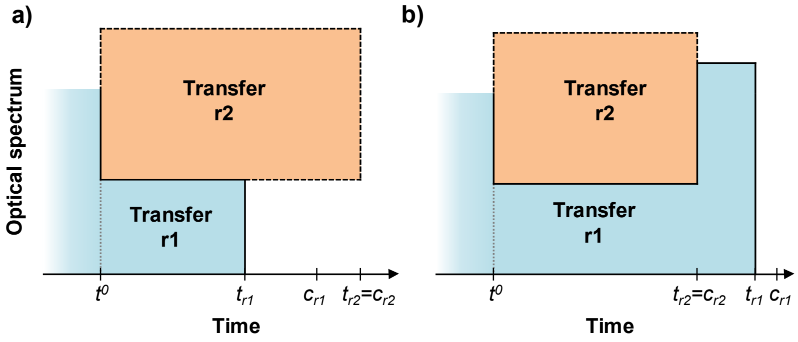

Figure 2. Scheduling schemes for spectrum allocation.

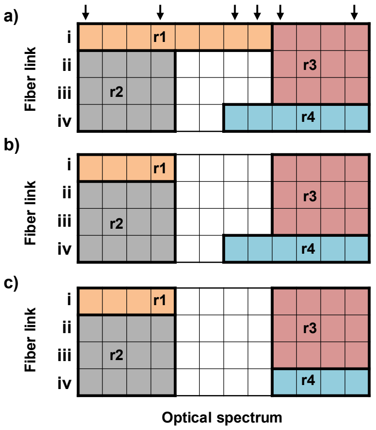

Figure 3. Spectral resources occupancy in a given route.

Table 1. Algorithm for RSSA.

```
INPUT: $p$, $b$, $br$, $sTED$
OUTPUT: $slot$
sw ← getSlotWidth ($p.length$, $br$)
D←getConnections($p$,$b$)
doSlot($D$, $sTED$)
S ← getFreeSlots($p$, $sTED$)
sort($S$, slot width, DESC)
for each $s \in S$ do
    Q ← getAdjacentPaths($p$, $s$, $sTED$)
    R ← findSlot($Q$, $s$, $sw$, $sTED$)
    if $R \neq \emptyset$ then
        doReschedule ($R$)
        slot ← getSA($p$, $br$, $sTED$)
        return $slot$
return NO_FEASIBLE_SOLUTION
```

Figure 3 illustrates an example of use and the solution returned by the algorithm. In Figure 3a, a new transfer request requiring a minimum bitrate equivalent to a slot of 4 frequency slices would be blocked in the route i-ii-iii-iv since no continuous slot is available. Before calling the RSSA algorithm, the feasible border slices (marked with an arrow in the figure) of currently established connections are obtained. When running the RSSA algorithm and evaluating the 4th border slice, i.e. higher frequency border slice in use by r1, the algorithm determines that r1 can be squeezed 4 slices to the left without exceeding its committed completion time. Then, a continuous slot of 2 slices is generated (Figure 3b). After that, the algorithm will try to obtain the required slot width (4 slices) by trying to squeeze the adjacent connections to the generated slot (r1 and r2 in the lower frequencies and r3 and r4 in the higher frequencies). Finally, the algorithm concludes that r4 can be squeezed and the desired slot width is found (Figure 3c).

3. ILLUSTRATIVE RESULTS

For evaluation purposes, the ability to deal with transfer-mode connection requests of the RSSA algorithm was compared against that of the RSA; i.e. without considering scheduled spectrum allocation. Three realistic national network topologies have been considered: the 21-node Telefonica (TEL), the 20-node British Telecom (BT), and the 21-node Deutsche Telekom (DT) networks. The optical spectrum width is fixed to 4 THz in each link, the slice width to 12.5 GHz with granularity 6.25 GHz, and all connections use the QPSK modulation format.

Transfer-mode offered load is related to VM migration among DCs. The amount of VMs to migrate between two DCs is proportional to the number of considered VMs and randomized using a uniform probability distribution function. We assume that transfer-mode requests are generated asking for completing the transference in 30 min. If the request cannot be served, one retrial is performed requesting resources so the committed completion time to be at most 1 hour. If the request cannot be served, it is finally blocked. Different values of the offered network load are created by changing the total amount of VMs. VMs size is set to 5.0 GB. Data volume is translated into required network throughput considering TCP, IP, Ethernet and MPLS headers.

Figure 4a shows the percentage of unserved bitrate for different offered loads and for the TEL and DT network topologies. For the sake of comparability between topologies, offered loads have been normalized to the value of the highest load. As summarized in Table 2, the proposed RSSA algorithm is able to serve more traffic in all three network topologies compared with using just RSA; i.e. from 12% to more than 30%. Interestingly, gain from using the RSSA is small under low loads as a result of enough resources are available to serve all the requests. As soon as the load increases, the ability of RSSA to perform elastic operations on the established optical connections, together with the scheduling to ensure that the committed completion time is not exceeded, brings clear benefits. Obviously, for high loads those benefits would cancel as few opportunities to reduce resources would exist. If we focus on 1% of unserved bitrate, we observe that noticeable gains in the offered load are obtained.

Regarding transference completion time, Figure 4b plots maximum and average times for both algorithms. Regarding average completion times, both algorithms perform similarly under low loads, whilst RSSA provides slightly higher times under high loads. The difference is amplified when we focus on maximum values, where completion time is pushed upwards to the maximum acceptable time (1 h) under high loads, so as to accept more incoming requests. In contrast, the RSA algorithm provides a linear increment as a function of the load, both for average and maximum values. Table 2 presents obtained completion times for the load unleashing 1% of unserved bitrate. It is worth noting that, although transference completion time is higher when the RSSA algorithm is used, the committed times were satisfied in all cases, demonstrating that the gains in load increment were not at the expense of failing to fulfil the committed completion time.

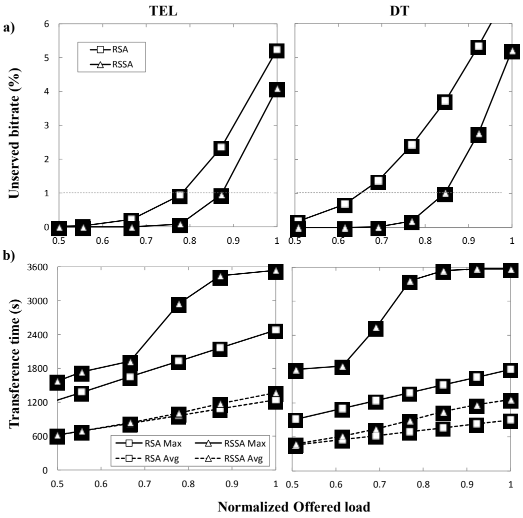

Figure 4. Unserved bitrate (a) and transference time (b).

Table 2. Offered load gain and transference time.

		TEL	BT	DT
Gain at 1% unserved bitrate		12.2%	17.8%	30.9%
Transference time in seconds (RSSA)	Max	3,444	3,523	3,543
	Avg	1,180	1,092	1,048
Transference time in seconds (RSA)	Max	2,167	1,822	1,518
	Avg	1,085	912	761

Table 3. Elastic operations at 1% of unserved bitrate.

		TEL	BT	DT
# Spectrum reductions	Max	3.5	3.6	3.9
	Avg	1.09	1.13	1.18
# Spectrum expansions	Max	4.1	4.8	5.6
	Avg	1.18	1.27	1.37
% of connections experiencing elastic ops.	Red.	9.3	18.8	24.4
	Exp.	23.6	43.5	53.4

Because the number of elastic operations performed on each connection is an important factor, Table 3 summarizes the number of elastic operations (spectrum reduction and expansion) performed over each established connection supporting a transference. The percentage of connections experiencing elastic operations is also summarized.

The average number of spectrum reductions and spectrum expansions that each connection experiences is just over one and arrives to a maximum of 4 reductions and 6 expansions. However, if we analyse the percentage of connections experiencing elastic operations, we realize that the percentage of expansions doubles that of spectrum reductions. Recall that spectrum reduction can be performed when a connection request arrives, whereas spectrum expansions are performed when connections are torn down, to implement scheduled allocations and to reduce transference completion time as described in the previous section.

It is clear that scheduled allocations are as a result of previous spectrum reduction, and thus the number of reductions should be greater or equal than the number of expansions. Therefore, in view of Table 3, the resources assignment policy when a connection is torn down is responsible for performing extra spectrum expansions to reduce transference completion time and increase resource utilization.

4. CONCLUSIONS

Transfer-mode requests and the routing and scheduled spectrum allocation problem were proposed in our previous work as a way to liberate the cloud from dealing with network specifics and complexity and manage effectively transfer-based requests respectively.

We extended that work and described the use of the proposed RSSA in scenarios where spectrum resources are highly utilized reducing thus spectrum fragmentation and chances to find continuous slots in a given route.

Illustrative results obtained by simulation showed remarkable gains in the offered load, as high as 30 %, when the RSSA algorithm was compared against that of the RSA for managing transfer-based datacentre connections.

REFERENCES

[1] L. Velasco, A. Asensio, J. Ll. Berral, E. Bonetto, F. Musumeci, and V. López, “Elastic operations in federated datacenters for performance and cost optimization,” *Elsevier Computer Communications*, vol. 50, pp. 142-151, 2014.

[2] A. Asensio, M. Ruiz, and L. Velasco, “Orchestrating connectivity services to support elastic operations in datacenter federations,” accepted in *Springer Photonic Network Communications*, 2015.

[3] D. King and A. Farrel, “A PCE-Based Architecture for Application-Based Network Operations,” IETF RFC 7491, Mar. 2015.

[4] L. Velasco, D. King, O. Gerstel, R. Casellas, A. Castro, and V. López, “In-operation network planning,” *IEEE Communications Magazine*, vol. 52, pp. 52-60, 2014.

[5] L. Velasco, A. Asensio, J.Ll. Berral, V. López, D. Carrera, A. Castro, and J.P. Fernández-Palacios, “Cross-stratum orchestration and flexgrid optical networks for datacenter federations*,” IEEE Network Magazine*, vol. 27, pp. 23-30, 2013.

[6] L. Velasco, A. Asensio, J.Ll. Berral, A. Castro, and V. López, “Towards a carrier SDN: An example for elastic inter-datacenter connectivity,” (Invited Paper) *OSA Optics Express*, vol. 22, pp. 55-61, 2014.

[7] A. Asensio, Ll. Gifre, M. Ruiz, and L. Velasco, “Carrier SDN for flexgrid-based inter-datacenter connectivity,” in *Proc. IEEE International Conference on Transparent Optical Networks (ICTON)*, 2014.

[8] A. Asensio and L. Velasco, “Managing transfer-based datacenter connections,” *IEEE/OSA Journal of Optical Communications and Networking (JOCN)*, vol. 6, pp. 660-669, 2014.