Joint Defragmentation of Spectrum and Computing Resources in Inter-Datacenter Networks over Elastic Optical Infrastructure

Xiahe Liu, Liang Zhang, Mingyang Zhang, Zuqing Zhu[†]
School of Information Science and Technology
University of Science and Technology of China, Hefei, China
[†]Email: {zqzhu}@ieee.org

***Abstract*—In this paper, we consider to apply a joint defragmentation (DF) of spectrum and computing resources in the inter-datacenter (inter-DC) networks built over elastic optical infrastructure based on the optical orthogonal frequency-division multiplexing (O-OFDM) technology. We propose joint DF algorithms that consist of a request selection strategy to select active requests for reconfiguring and an anycast-based DF routing and spectrum assignment (DF-RSA) to reconfigure the selected requests. For the joint DF, two request selection strategies and three anycast-based DF-RSA schemes are designed, and their combinations are evaluated with simulations. The simulation results show that by considering the usages of spectrum and computing resources jointly, the proposed algorithms can improve the blocking performance of the inter-DC network significantly with controlled numbers of DF operations.**

***Index Terms*—Inter-datacenter networks, Anycast, Elastic optical networks (EONs), Resource defragmentation.**

I. Introduction

Nowadays, with the rapid development of cloud computing, datacenter (DC) networks have attracted numerous attentions. Since the traffic patterns in DC networks exhibit the co-existence of huge peak throughput and high burstiness [1], efficient support from the physical infrastructure is highly desired. It is known that since optical fibers have tremendous bandwidth, optical networking can provide a viable infrastructure solution to DC networks, especially for the inter-DC networks [2]. Moreover, recent advances on the optical orthogonal frequency-division multiplexing (O-OFDM) technology have shown that over Tb/s transmission capacity and flexible bandwidth adjustment with a small granularity of 12.5 GHz can be achieved simultaneously [3]. Therefore, the elastic optical infrastructure based on O-OFDM becomes a promising candidate to support the inter-DC networks, as it enables agile spectrum management and facilitates seamless integration of optical transmission and upper-layer applications [4].

Besides the aforementioned advantages, O-OFDM networks also bring in challenges, as for bandwidth allocation, network operators need to manipulate blocks of contiguous subcarrier frequency slots (FS') instead of discrete wavelength channels [3, 4]. More importantly, spectrum fragmentation, *i.e.*, the existing of non-aligned, isolated, and small-sized unused FS blocks in optical spectra, also becomes a serious problem and can degrade network performance significantly [5]. Previously, people have proposed spectrum defragmentation (DF) approaches that could reduce spectrum fragmentation with network reconfigurations [5–7]. Note that in addition to bandwidth requirement, the requests in an inter-DC network usually also associate with computing demands that need to be executed on virtual machines (VMs) in destination DCs [8]. Hence, if the computing loads are not distributed properly among DCs, we may have the fragmentation of computing resources, which also degrades the network performance. In [9], Sridharan *et al.* investigated computing resource defragmentation in virtual desktop clouds. One attractive attribute of applications such as cloud computing in inter-DC networks is that the destination DC of a request can be flexible. To this end, we expect that by leveraging network reconfiguration with anycast [10], the spectrum resources on fibers and the computing resources in DCs can be utilized more wisely [11].

In this paper, to the best of our knowledge, we consider to apply joint defragmentation (DF) of spectrum and computing resources in inter-DC networks built over elastic optical infrastructure based on O-OFDM for the first time. More specifically, in order to improve the blocking performance of an inter-DC network, we optimize the utilizations of the spectrum and computing resources jointly with complexity-controlled network reconfigurations. Note that the network reconfiguration can change not only the routing and spectrum assignment (RSA) but also the destination DC of a request. The RSA change can be done with physical-layer techniques, such as the hop-tuning based retuning [12], while the destination change can be done with DC service migration. The rest of the paper is organized as follows. We formulate the problem of joint DF of spectrum and computing resources in inter-DC networks in Section II. Section III discusses the proposed algorithms for joint DF. Simulation results for performance evaluation are shown in Section IV. Finally, Section V summarizes the paper.

II. Problem Formulation

Fig. 1 shows the examples of spectrum DF and joint DF. It can be seen that with spectrum DF only, we consolidate the spectrum usage on fiber links, *i.e.*, the bandwidth resources on *Path* 1 is reconfigured. However, as the destination DC is unchanged, and usage of computing resources on the DCs

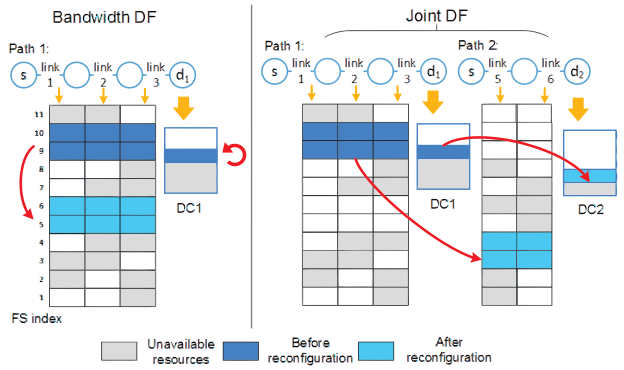

Fig. 1. Examples of spectrum DF and joint DF.

are not re-optimized. While with joint DF, we rearrange the utilization of both resources, and another feasible DC can be chosen as the new destination by leveraging the property of anycast. For instance, the joint DF in Fig. 1 selects *DC* 2 as the new destination, and the bandwidth and computing resource utilizations migrate from *Path* 1 to *Path* 2 and from *DC* 1 to *DC* 2, respectively. Therefore, we can make the resources better utilized with joint DF.

A. Network Model and Request Model

The physical topology of the inter-DC network is modeled as a graph $G(V, E)$, where V is the set of nodes and E represents the set of the fiber links. We consider the situation that the DCs are located near the switching nodes, and thus each node $v \in V$ associates with an available DC computing capacity denoted as C_v. Based on the working principle of O-OFDM, we assume that there are B subcarrier frequency slots (FS') to allocate on each link $e \in E$. In the dynamic network scenario, a request is denoted as $R(s, d, r_b, r_c, t_a, t_h)$, where s and d are the source and destination nodes, r_b is the bandwidth demand in number of FS', and t_a and t_h are the arrival time and hold period, respectively. For the computing demand of the request r_c, we assume that it scales linearly with the bandwidth demand as [8]

$$r_c = \alpha \cdot r_b, \tag{1}$$

where α is the scaling constant. For simplicity, we assume that the lightpath of a request is always set up all-optically end-to-end over a single routing-path, under the spectrum contiguous, non-overlapping and continuity constraints [5].

B. Request Serving

A request $R(s, d, r_b, r_c, t_a, t_h)$ is initially served with the K-shortest paths and balanced load spectrum assignment (KSP-BLSA) algorithm [13]. If the bandwidth demand r_b can be provisioned with the spectrum resources on the selected routing path and the computing demand r_c satisfies $r_c \leq C_d$, the request is considered as successfully served, otherwise, it is blocked. Blocking probability is one of the most important metrics to evaluate the network performance. Since different requests may have different bandwidth demands, we define the bandwidth blocking probability (BBP) for fair comparisons

$$BBP = \lim_{T \to \infty} \frac{N_b(T)}{N(T)}, \tag{2}$$

where $N_b(T)$ and $N(T)$ are the blocked and total bandwidth demands during period $[0, T]$, respectively. Due to the fact that in this work, the computing demand scales linearly with the bandwidth demand, the blocking probability of computing demands should be the same as BBP.

C. Joint Defragmentation

Setting up and tearing down requests dynamically in the inter-DC networks can result in resource fragmentation on fiber spectra and DC computing capacities. To avoid invoking the DF operations too frequently, we only consider the responsive approach. Specifically, we invoke a DF operation whenever a request would be blocked otherwise. In each DF operation, a network-wide partial reconfiguration is performed to consolidate the utilizations of spectrum and computing resources. Here, we invoke "network-wide" reconfiguration to ensure fairness in the network. Otherwise, if we perform local reconfiguration to accommodate a particular request, that request is treated with an unfairly high priority. Moreover, "partial" reconfiguration means that the DF operation only reallocates the spectrum and computing resources for a portion of the active requests. In the network, if the set of active requests is denoted as $\mathbb{R}$, we select $\beta \cdot |\mathbb{R}|$ requests to reconfigure in each DF operation. Here, $|\mathbb{R}|$ returns the number of the active requests, and $\beta \in (0, 1)$ is the selection ratio.

III. Joint Defragmentation Algorithms

Basically, each DF operation consists of two steps: 1) selecting certain active requests to reconfigure, and 2) reconfiguring the selected requests with an anycast-based defragmentation RSA (DF-RSA). *Algorithm* 1 shows the overall procedures of the joint DF algorithm. The request selection strategy and the anycast-based DF-RSA are discussed below.

A. Request Selection Strategies

In this subsection, we discuss two request selection strategies for selecting the active requests to reconfigure.

1) Highest Used FS-Index First (HUSIF) Strategy: This strategy selects requests based on the indices of their assigned FS' [5]. Specifically, HUSIF chooses the first $\beta \cdot |\mathbb{R}|$ active requests whose highest used FS-indices are the largest. HUSIF is a spectrum-oriented selection strategy.

2) Highest Computing Load First (HCLF) Strategy: This strategy selects requests based on the computing loads of their destination DCs, and is a computing-oriented selection strategy. *Algorithm* 2 shows the detailed procedures. We denote the average available computing capacity of all the DC as $\overline{C}$. In the loop from *Line* 2 to 13, we check the DC nodes from the busiest to the idlest, based on their available computing capacities. Then, in *Lines* 3-9, we select the active requests on the current DC in the ascending order of their

Algorithm 1: Overall Joint Defragmentation

while *the inter-DC network is operational* **do**
  release resources of expired requests;
  try to serve a pending request with KSP-BLSA;
  if *the request will be blocked* **then**
    select $\beta \cdot |\mathbb{R}|$ active requests for reconfiguration based on a selection strategy;
    re-optimize resource allocations of the selected requests with an anycast-based DF-RSA;
    migrate the selected requests to new resource allocations;
  end
  try to serve the request again with KSP-BLSA;
  if *the request will still be blocked* **then**
    mark the request as blocked;
  end
end

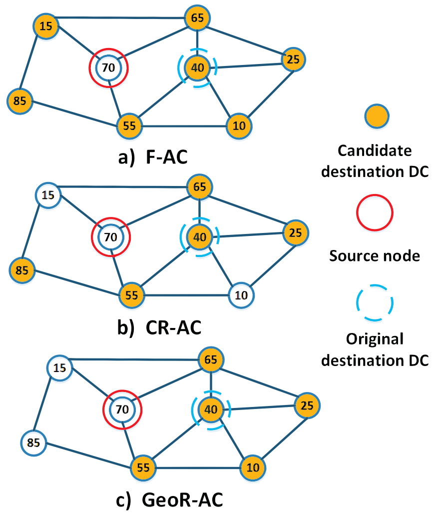

Fig. 2. Examples of destination selection in the anycast schemes.

computing demands and update the DC's available computing capacity C_v, until C_v is less than or equal to $\overline{C}$. Basically, we believe that the the computing demands of these requests are causing fragmentation on computing resources and should be reallocated for computing DF. The whole selection process stops when the preset quota, *i.e.*, $\beta \cdot |\mathbb{R}|$, has been reached, or the computing capacity of each DC is equalized.

Algorithm 2: Highest Computing Load First Strategy

input : Physical topology $G(V, E)$, active request set $\mathbb{R}$, and DF ratio β.
output: Active requests selected to reconfigure.

$j = 0$;
for *all* $v \in V$ *in ascending order of* C_v **do**
  for *all active requests destined to* v *in ascending order of* r_c **do**
    mark the request as selected to reconfigure;
    $j = j + 1$;
    if $j = \beta \cdot |\mathbb{R}|$ *or* $C_v \leq \overline{C}$ **then**
      break;
    end
  end
  if $j = \beta \cdot |\mathbb{R}|$ *or* C_v *is equalized* $\forall v \in V$ **then**
    break;
  end
end

B. Anycast-based DF-RSA

In this subsection, we develop an anycast-based DF-RSA algorithm to reconfigure the selected requests for joint DF. *Algorithm* 3 shows the proposed algorithm in detail.

The algorithm operates on a candidate destination node set $V_p \subseteq (V \setminus \{s\})$ for a selected request $R(s, d, r_b, r_c, t_a, t_h)$. In this work, we consider three anycast schemes, *i.e.*, full anycast (F-AC), computing restricted anycast (CR-AC), and geographical restricted anycast (GeoR-AC). For F-AC, there is no constraint on where the destination DC of a selected request can be changed to in the anycast-based DF-RSA, and thus we have $V_p = V \setminus \{s\}$. However, as we will show later in the performance evaluation, when being used together with the HUSIF request selection strategy, F-AC may cause the computing loads to be unevenly distributed in the network and can only provide limited BBP performance improvement. Therefore, in CR-AC, we add a constraint to ensure that the destinations of the selected requests will only be migrated to the DCs whose available computing capacities are larger than or equal to certain portion of the average value $\overline{C}$, *i.e.*, $V_p = \{v : C_v \geq \gamma \cdot \overline{C}\} \setminus \{s\}$. Here, $\gamma \in (0, 1)$ is a preset constant. GeoR-AC is a more practical anycast scheme with the consideration of service migration delay, in which we only allow the request's destination DC to be changed to its adjacent nodes or itself, *i.e.*, $V_p = \{d, \{v : (d, v) \in E\}\} \setminus \{s\}$. Therefore, we can avoid the cases in which the service migration encounters too much delay and causes prolonged service disruption. Fig. 2 illustrates intuitive examples of the destination selection in the three anycast schemes. The numbers on the nodes are their available computing capacity, and for CR-AC, we set $\gamma = \frac{1}{3}$.

In *Algorithm* 3, *Lines* 1-6 are for initialization and ignore the DC nodes whose available computing capacities are insufficient. The anycast-based DF-RSA is conducted in *Lines* 7-18. Basically, we perform the fragmentation-aware RSA (FMA-RSA) developed in [14] to obtain the new resource allocation, *i.e.*, destination DC and RSA, for each selected request.

IV. Performance Evaluations

The proposed joint DF algorithms are evaluated with simulations using the NSFNET topology shown in Fig. 3. We assume that each fiber link accommodates $B = 358$ FS'. Each node in the topology connects to a DC locally, and

Algorithm 3: Anycast-Based DF-RSA

input : Physical topology $G(V, E)$, and an active request $R(s, d, r_b, r_c, t_a, t_h)$.
output: New resource allocation for the request.

```
V_p = ∅, cost_min = +∞;
for all v ∈ V \ {s} do
    if C_v > r_c and v satisfies the anycast scheme
    (F-AC or CR-AC or GeoR-AC) then
        V_p = V_p ⋃ {v};
    end
end
for all v ∈ V_p do
    try to perform FMA-RSA in [14] for R with s-v as
    the source-destination pair;
    if FMA-RSA succeeds with s-v then
        calculate the fragmentation-aware cost [14] for
        the RSA solution;
        store the cost in cost;
        if cost < cost_min then
            cost_min = cost;
            delete previous new resource allocation;
            store current new resource allocation;
        end
    end
end
return the stored new resource allocation;
```

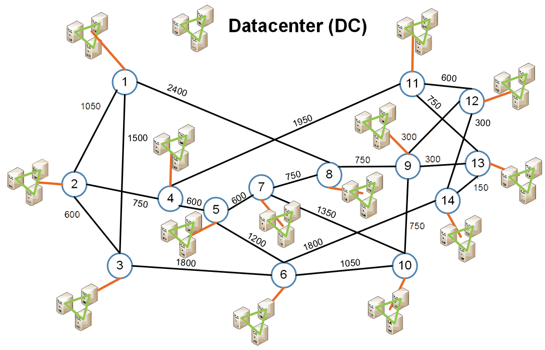

Fig. 3. The inter-DC network topology (link lengths in kilometers).

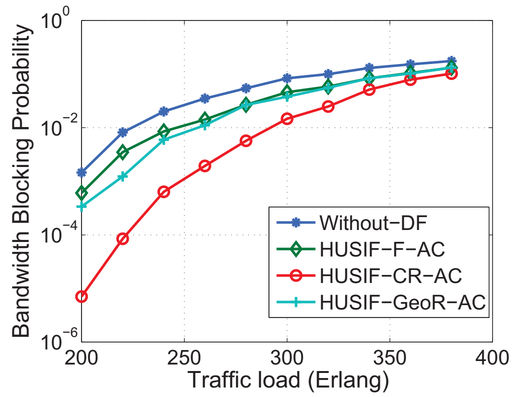

Fig. 4. BBP results of HUSIF based joint DF algorithms.

when the inter-DC network is empty, the available computing capacities are identical on all the nodes as $C_v = 640$ units after normalization. For a dynamic request $R(s, d, r_b, r_c, t_a, t_h)$, its source and desired destination nodes (s and d) are randomly selected, the bandwidth demand r_b is uniformly distributed in $[1, 16]$ FS', and the computing demand is calculated as $r_c = \alpha \cdot r_b$ with $\alpha = 0.2$. The dynamic requests' arrivals follow the Poisson process with the average arrival rate as λ, and the holding time t_h of each request follows the negative exponential distribution with the mean value $\frac{1}{\mu}$. Hence, we can quantify the traffic load as $\frac{\lambda}{\mu}$ in Erlangs. In the simulations, we set the selection ratio as $\beta = 0.3$ and the constant γ in CR-AC as $\gamma = \frac{1}{3}$, for each of the varies traffic load, the total requests number is about 10 thousand. We name the proposed algorithms with the combined abbreviations of the request selection strategy and anycast scheme. For instance, "*HUSIF-CR-AC*" denotes the joint DF algorithm that adopts HUSIF request selection strategy and CR-AC anycast scheme.

Fig. 4 shows the simulation results on BBP from the algorithms that use HUSIF as the request selection strategy. We observe that compared with the scheme without DF, the joint DF algorithms can improve the BBP performance no matter what anycast scheme is used. It is interesting to notice that the BBP performance of *HUSIF-F-AC* is the worst among in Fig. 4. In the inter-DC network, there are three scenarios that lead to a request blocking, *i.e.*, spectrum blocking, computing blocking, and combined blocking. Spectrum blocking refers to the scenario that there is enough computing resource on the destination d but the spectrum resource on the routing paths between the s-d pair is insufficient. If the situation is opposite, the scenario is computing blocking. If both resources are insufficient, the scenario is combined blocking. Since HUSIF does not consider computing loads on the DCs and the consequent F-AC may migrate requests to the DCs whose computing loads are already fairly high, *HUSIF-F-AC* causes uneven computing load distribution in the network and induce high computing blocking. Therefore, it only provides limited BBP performance improvement. This can be verified with the results on BBP break-downs in Table. I. It can be seen that when the traffic load is 360 Erlangs, *HUSIF-F-AC* has much higher computing blocking than *HUSIF-CR-AC*. For *HUSIF-CR-AC*, the usages of both the spectrum and computing resources are addressed properly and hence it can reduce BBP more. When we have to control the service migration delay, *HUSIF-GeoR-AC* restricts the destination selection in anycast and therefore its results on BBP are larger than those of *HUSIF-CR-AC*. However, since GeoR-AC restricts where the destination DCs can be migrated to, which to certain extent also prevents the generation of computing "hot-spots", its BBP results are slightly lower than those of *HUSIF-F-AC*.

Fig. 5 illustrates the BBP results from the algorithms that use HCLF as the request selection strategy. It can be seen that

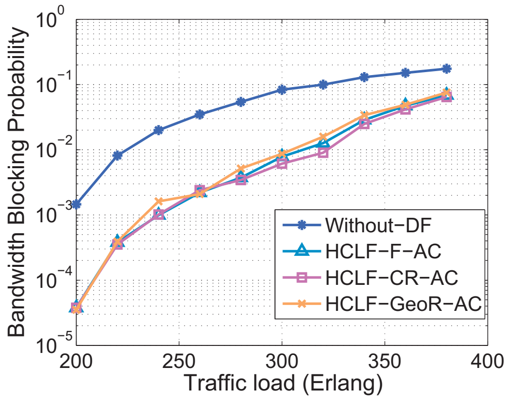

Fig. 5. BBP results of HCLF based joint DF algorithms.

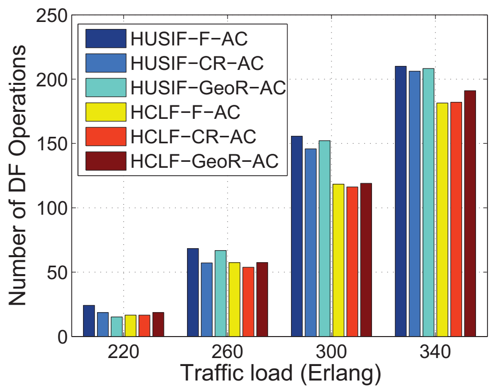

Fig. 6. Results on the numbers of DF operations.

all three DF algorithms achieve significant reductions on BBP and their performance is similar. Since HCLF considers the computing resources while the consequent anycast-based DF-RSA takes spectrum resources into account, the fragmentation on both the spectrum and computing resources are addressed properly in the joint DF. This explanation can also be verified with the results in Table. I, in which we can see that compared with the HUSIF based ones, the HCLF based algorithms achieve much lower computing blocking by sacrificing little performance on the spectrum blocking and provide lower overall BBP. Meanwhile, since HCLF has already considered computing resource usage, the BBP differences caused by the anycast schemes are also reduced significantly. The BBP results from *HCLF-CR-AC* are still slightly lower than those from *HCLF-F-AC* and *HCLF-GeoR-AC*.

TABLE I
BBP BREAK-DOWN FOR THREE BLOCKING SCENARIOS (360 ERLANGS)

	Overall	Spectrum Blocking	Computing Blocking	Combined Blocking
HUSIF-F-AC	10.45%	0.00%	10.45%	0.00%
HUSIF-CR-AC	7.85%	0.00%	7.85%	0.00%
HUSIF-GeoR-AC	10.25%	0.00%	10.25%	0.00%
HCLF-F-AC	4.74%	0.62%	4.05%	0.07%
HCLF-CR-AC	4.16%	0.72%	3.39%	0.05%
HCLF-GeoR-AC	4.90%	1.07%	3.71%	0.12%

When we compare the BBP results in Figs. 4 and 5, we observe that the HCLF based algorithms provide lower BBP results than *HUSIF-F-AC* and *HUSIF-GeoR-AC*. Figs. 4 and 5 also show that *HUSIF-CR-AC* achieves the best BBP performance for low traffic loads ($\leq$ 260 Erlangs), while its BBP performance is worse than that of the HCLF based algorithms for the traffic loads higher than 260 Erlangs. Note that when the traffic load is low, the computing resources are sufficient and we should concentrate on spectrum DF to reduce BBP. We believe this is the reason why *HUSIF-CR-AC* can achieve the best BBP performance for low traffic loads.

For each traffic load, the results on the numbers of DF operations from different joint DF algorithms are plotted in Fig. 6. We can see that the HCLF based algorithms typically invokes less DF operations than the HUSIF based ones.

V. CONCLUSION

We investigated the joint DF of spectrum and computing resources in the inter-DC networks over elastic optical infrastructure. The proposed algorithms were based on a request selection strategy and an anycast-based DF-RSA. We designed two request selection strategies and three anycast-based DF-RSA schemes, and evaluated their combinations with simulations. The simulation results showed that by addressing the fragmentations of the spectrum and computing resources jointly, the proposed algorithms could improve the blocking performance of the inter-DC network with controlled numbers of DF operations.

ACKNOWLEDGMENTS

This work was supported in part by the Program for New Century Excellent Talents in University (NCET) under Project NCET-11-0884, the National Natural Science Foundation of China (NSFC) under Project 61371117, the Fundamental Research Funds for the Central Universities (WK2100060010), and the Strategic Priority Research Program of the Chinese Academy of Sciences (XDA06010302).

REFERENCES

[1] C. Liu, A. Kind, and A. Vasilakos, “Sketching the data center nework traffic,” *IEEE Network*, vol. 27, pp. 33–39, Jul./Aug. 2013.

[2] C. Develder *et al.*, “Optical networks for grid and cloud computing applications,” *Proc. IEEE*, vol. 100, pp. 1149–1167, May. 2012.

[3] O. Gerstel *et al.*, “Elastic optical networking: a new dawn for the optical layer?” *IEEE Commun. Mag.*, vol. 50, pp. S12–S20, Apr. 2012.

[4] B. Yoo, Y. Yin, and K. Wen, “Intra and inter datacenter networking: The role of optical packet switching and flexible bandwidth optical networking,” in *Proc. of ONDM 2012*, pp. 1–6, Apr. 2012.

[5] M. Zhang *et al.*, “Bandwidth defragmentation in dynamic elastic optical networks with minimum traffic disruptions,” in *Proc. of ICC 2013*, pp. 1–5, Jun. 2013.

[6] F. Cugini *et al.*, “Push-pull defragmentation without traffic disruption in flexible grid optical networks,” *J. Lightw. Technol.*, vol. 31, pp. 125–133, Jan. 2013.

[7] R. Wang and B. Mukherjee, “Provisioning in elastic optical networks with non-disruptive defragmentation,” *J. Lightw. Technol.*, vol. 31, pp. 2491–2500, Aug. 2013.

[8] J. Buysse *et al.*, "Energy-efficient resource-provisioning algorithms for optical clouds," *J. Opt. Commun. Netw.*, vol. 5, pp. 226–239, May. 2013.

[9] M. Sridharan *et al.*, "Defragmentation of resources in virtual desktop clouds for cost-aware utility-optimal allocation," in *Proc. of UCC 2011*, pp. 253–260, Dec. 2011.

[10] W. Jia, D. Xuan, and W. Zhao, "Integrated routing algorithms for anycast messages," *IEEE Commun. Mag.*, vol. 38, pp. 48–53, Jan. 2000.

[11] M. Gharbaoui, B. Martini, and P. Castoldi, "Anycast-based optimizations for inter-data-center interconnections," *J. Opt. Commun. Netw.*, vol. 4, pp. B168–B178, Nov. 2012.

[12] R. Proietti *et al.*, "Rapid and complete hitless defragmentation method using a coherent rx lo with fast wavelength tracking in elastic optical networks," *Opt. Express*, vol. 20, pp. 26 958–26 968, Nov. 2012.

[13] Y. Wang, X. Cao, and Y. Pan, "A study of the routing and spectrum allocation in spectrum-sliced elastic optical path networks," in *Proc. of INFOCOM 2011*, pp. 1503–1511, Apr. 2011.

[14] Y. Yin *et al.*, "Fragmentation-aware routing, modulation and spectrum assignment algorithms in elastic optical networks," in *Proc. of OFC 2013*, pp. 1–3, Mar. 2013.