Received: 1 November 2019 | Revised: 20 January 2020 | Accepted: 30 January 2020
DOI: 10.1002/dac.4360

RESEARCH ARTICLE

WILEY

Hop distance–based bandwidth allocation technique for elastic optical networks

Stathis B. Mavridopoulos[1] | **Georgia Beletsioti**[1] | **Petros Nicopolitidis**[1] | **Georgios I. Papadimitriou**[1] | **Emmanouel Varvarigos**[2]

[1]Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

[2]School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece

Correspondence
Petros Nicopolitidis, Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece.
Email: petros@csd.auth.gr

Funding information
Operational Program Competitiveness, Entrepreneurship and Innovation, Grant/Award Number: T1EDK-05061

Summary

Elastic optical networks (EON) have emerged as a solution to the growing needs of the future internet, by allowing for greater flexibility, spectrum efficiency, and scalability, when compared to WDM solutions. EONs achieve those improvements through finer spectrum allocation granularity. However, due to the continuity and contiguity constrains, distant connections that are routed through multiple hops suffer from increased bandwidth blocking probability (BBP), while more direct connections are easier to form. This paper proposes HopWindows, a novel method that strategically allocates bandwidth to connections based on their hop distance. This new algorithm applies masks that control the range of frequency slots (FSs) allocated to each n-hop connection. Furthermore, a new network metric is introduced, the normalized bandwidth blocking probability (normalized BBP). Utilization of this metric ensures increased fairness to distant connections. Extended simulation results are presented which indicate that the proposed HopWindows method achieves a superior performance over the well-known FirstFit algorithm. The proposed algorithm may achieve a decrease in bandwidth blocking probability of up to 50%.

KEYWORDS

bandwidth allocation, blocking probability, elastic networks, RSA

1 | INTRODUCTION

Elastic optical networks (EONs) offer flexibility in how to allocate just the appropriate bandwidth capacity to the connections and are considered the network solution for backbone and next generation metropolitan networks. In flexible grid technologies, the spectrum is split into slots of 6.75, 12.5, or 25 GHz. These frequency slots (FSs) are then combined to create channels that do not overlap due to OFDM's orthogonality,[1] in order to serve custom sized bandwidth requests and adapt to the dynamic and heterogeneous nature of their arrivals during network's operation.

The main focus of routing and spectrum allocation assignment (RSA) EON algorithms is that connections must satisfy both the continuity and frequency contiguity constraints.[2,3]

Our work addresses the fairness problem that multihop connections face, specifically that connections comprising many hops suffer from a larger BP compared to those over fewer hops, as satisfying the continuity and contiguity constraint over longer distances becomes increasingly harder. Thus, we propose a new metric in this paper, the *normalized* *to hop-distance bandwidth blocking probability* (*normalized_BBP*) that considers connection hop distance and its effect on network performance.

A similar fairness issue is highlighted in Rosa et al,[4] where it is shown that bandwidth-consuming connections are characterized by higher blocking probability. Finding larger continuous groups of unoccupied FSs is hard; contiguity exacerbates the situation. Light bandwidth requests are more probable to fit in the spectrum and experience lower probability of blockage. Another work that takes interest into connection distance is Chatterjee and Oki.[5] Low-index FSs are affected by dispersion less than high-index ones, meaning that long-distance lightpaths suffer more quality-of-transmission degradation when placed in a high-index FS. Considering this phenomenon, the number of FSs per connection can be minimized, leading to improvements for the network performance. Our work takes a different approach by utilizing connection hop distance instead of lightpath length.

A low complexity, effective RSA solution that is used in many works is the FirstFit (FF) algorithm.[6,7] Comellas and Junyent[8] improve upon FF by limiting the allowed connections' bandwidth size. Connections are allowed to occupy only 2, 5, or 8 FSs (2-5-8 traffic load profile), thus reducing the link fragmentation. Our proposed method, called HopWindows (HW), can be employed using any traffic load profile and improves both upon FF and 2-5-8 scheme by taking hop distance into consideration.

This work extends upon Mavridopoulos et al.[9] The work in Mavridopoulos et al[9] is limited by its hard set of the path utilization limit to 50%, that we now replace with the *activate* parameter, which is calculated in Algorithm 2. Performance results show a reduction up to 50% in BBP.

2 | PROPOSED SCHEME

2.1 | Normalized BBP

One of the most commonly used metrics employed when investigating the performance of EONs is bandwidth blocking probability (BBP) and is calculated as

$$BBP = \frac{\sum bandwidth\_blocked}{\sum total\_bandwidth\_generated}. \tag{1}$$

This metric is undeniably useful but does not reflect the total reality, especially concerning connections comprising many hops. Depending on the underlying routing algorithm, one or more paths are available between a pair of nodes.

Minimum hop distance (*conn_min_hops*) is an important parameter in our method and refers to the minimum hop length of all allowed paths between two nodes. Neighbor nodes have a minimum hop distance of one.

EONs allow great flexibility and increased bandwidth utilization; however, connections are required to satisfy both the continuity and contiguity frequency constraints. Since high hop connections require free overlapping FSs over multiple links, they suffer from higher blocking probability. Our proposed method introduces a new metric called *normalized_bpp* and is calculated as

$$normalized\_BBP = \frac{\sum_{i=1}^{C} bandwidth\_blocked_i \cdot min\_hops_i}{\sum_{i=1}^{C} bandwidth\_generated_i \cdot min\_hops_i} \tag{2}$$

for each of the C connections generated in the network.

In *normalized_BBP*, bandwidth block probability is weighted against the minimum connection distance. The logic behind this parameter is that long hop-distance connections occupy more FSs in more links, when compared to more direct connections. If network performance is examined only against BBP, then a scheme that bans all long connections would superficially perform better. In such a network, if offered load is sufficiently high, then multiple short connections could be established instead of a single long connection. The BBP performance would seemingly improve, while in reality, all possible connections comprising many hops are disabled.

Algorithm 1: HopWindows

```
New Connection

Input  : conn_min_hops: minimum number of hops for this connection
         Paths[i]: k3 shortest paths
         Paths[i][max_util]: Percentage of FSs in path that are occupied
         Paths[i][len]: number of hops for this path
         Paths[i][free_slots]: available FSs on path
         masks[conn_min_hops]: limit FS according to the mask
         activate: parameter computed in Alg.2
foreach path ∈ Paths do
    Find a path that fits this connection
    if path[max_util] > activate then
        This path is at risk of congestion, enable HW
        if path[len] > conn_min_hops + 1 then
            Do not use "detour" paths
            continue
        else
            apply masks[conn_min_hops] on path_free_slots
    if connection fits in path_free_slots then
        Apply firstfit for this path
        return path
Output: Select Path | Reject connection
```

Algorithm 2: Calculate masks and activate parameter

```
Input  : topology: contains the network topological information
         simulation(masks, activate, topology): will run a simulation of HW and returns the normalized_BBP
         Masks[conn_min_hops]: limit FS according to the mask
         max_fs: maximum FS index
masks[i] ← [1, max_fs]                                  ▷ neutral/default masks
activate ← max_fs − −                                   ▷ the Alg. 1 activate parameter
norm_BBP ← simulation(masks, topology)
while True do
    foreach mask ∈ Masks do
        while True do
            mask − −                 ▷ decrease mask → limit bandwidth for n-hops connections
            new_norm_BBP ← simulation(mask, activate, topology)
            if new_norm_BBP > norm_BBP then
                mask + +                                ▷ revert mask, no improvement
                break                                   ▷ continue to next mask
            else
                norm_BBP ← new_norm_BBP                 ▷ update to improved BBP
    if the foreach run once without improvement then
        if have not searched for new activate then
            activate − −                                ▷ decrease and test again
        else
            activate + +                                ▷ revert
            break
Output: The Masks[i] & activate parameter used in Alg.1
```

2.2 | HW scheme

The proposed HW algorithm is described in Algorithm 1. The HW algorithm utilizes the parameters *activate* and *masks*, which are calculated in Algorithm 2. The *activate* parameter controls when HW is enabled. Under low load conditions, the EON operates using simple FF.[6] When a new connection request is generated, the k shortest paths between source and destination are examined for unoccupied FSs that are continuous and contiguous. If enough FSs are available, then the connection can be committed, or else, the connection is blocked.

HW differentiates from FF in step 4 of Algorithm 1. Similar to how proactive defragmentation algorithms[10] operate, HW is enabled when path utilization is higher than the *activate* threshold. In that case, the algorithm first attempts to limit usage of highly occupied paths (steps 6-8). This check preemptively disables connections from establishing "detour/cyclic" paths and is integral part of our proposed method. A side benefit of this rule is that connections may select an alternative less occupied path, resulting in a rudimentary load balancing of the network links.

In the next step, the precalculated mask is applied to the profile of free FSs of the path. There are N masks used in our algorithm, where N is the minimum hop distance for the more distant source—destination pair in the network. For example, in the case of NSFnet used later in the Section 3, N is equal to 4, since the longest from the shortest paths in that topology consists of four hops.

Each mask comprises of two numbers that represent the range of FS available to the n-hop connection. For example, if the minimum hop distance for a connection is 3, then the mask $mask[3] = [0, 100]$ has the meaning that only FSs with index from 0 to 100 can used.

After the mask rule is applied to the $path[free\_slots]$, the algorithm attempts to place the connection to the leftmost slot of continuous FSs that can fit it and are allowed by the mask.

If the connection fails to fit, the algorithm examines the next available path or the connection is blocked.

2.3 | Precalculating HW parameters

HW Algorithm 1 operates using the *activate* and *masks* parameters. Our analysis through simulation demonstrates that both parameters are unique to each topology employed and are dependent on the traffic profile used. There is no generic rule set that can satisfy multiple and diverse scenarios, such as those examined in Section 3. Our suggestion to this problem is to diverge to a good solution by pre-processing.

Similarly to the training phase of machine learning algorithms,[11,12] Algorithm 2 searches for good solutions to the *activate* and *masks* parameters via simulation.

Firstly, a baseline is calculated by using "default" masks for all hop distances. There are N masks used, where N is the hop count of the longest shortest path in the network. The default masks are in the form of $[1, max\_fs]$ and do not restrict connections at all. Then, simulated runs of the network calculate *normalized_BBP* for each combination of the masks and the *activate* parameters. Gradually, *activate* decreases, meaning HW is activated more commonly and for less populated paths. The search does not stop while *normalized_bpp* keeps improving. The parameters used for each topology in Section 3 are calculated using this method.

3 | PERFORMANCE EVALUATION

The performance characteristics of HW were investigated through simulation for the 14-node 21-link NSFnet,[13] a mesh-based network,[14] which consists of 29 nodes and 41 links, and a ring topology of 16 nodes. NSFnet is commonly used as a testing topology, while the mesh-based and ring networks are selected as representatives of a metropolitan area network topology.

Each-way connected fiber has the capacity of a maximum of 160 FSs, a value which is typically used in previous works. The source and destination nodes of a traffic request are uniformly selected, and the traffic load is dynamically generated and calculated as λ / μ (Erlang). The interarrival connection rate λ is constant and equal to 1. The connection duration parameter μ follows a negative exponential distribution. The range of offered loads used is different in each scenario, since the maximum load capacity of each topology is different.

We tested for three connection load scenarios: *elastic*, the *4-7-12* rule, and the *2-5-8* rule. In the case of *elastic*, the connection load is in the range of $[2, 15]$ FSs, including guardband, while the rule *4-7-12* describes the scenario of three distinct services with requirements of 4 FSs, 7 FSs, and 12 FSs, including guardbands.[15] The *2-5-8* rule is found to improve FF performance by restricting the allowed connection size to 2, 5, or 8 FSs.[8]

Compared to the FF scheme, our method never underperforms. The *activate* parameter ensures that the worst case performance of HW is at least on par with the FF method, since HW is enabled only when there are benefits to be realized. Table 1 shows $bl(i)$, the BBP for the FF, and HW in NSFNet for the 2-5-8 load profile for connections comprising i hops. These results show that a significant decrease in $bl(i)$ for long connections ($i = 3, 4$ hops), which sometimes results only to a slight increase of BBP in short connections ($i = 1, 2$ hops).

The best performance improvements were observed for low offered loads in the NSF topology. This is because the NSF contains more bottlenecks, where most of the long hop connections need to pass. HW is often activated for those links, and this differentiation explains its superiority over FF. Specifically, at 150 Erlangs and for the *2-5-8* load profile, the *normalized_BBP* is cut in half (0.78% for HW and 1.5% for FF). In the case of *4-7-12*, there is a *normalized_BBP* reduction of 27.5%, from 8% to 5.8%, and in the case of the *elastic* load profile, the reduction is 22.5%, from 10.3% to 7.86%. Network performance for low offered loads is more interesting, since a real-world scenario will operate within those ranges of low BBP.

TABLE 1 BBP comparison for FF and HW for NSFnet and 2-5-8 load profile

Erlangs / i(hops)	150		208				325		383		441		500	
	FF	HW	FF	HW	FF	HW	FF	HW	FF	HW	FF	HW	FF	HW
1	0.16	0.14	0.68	0.72	1.52	1.85	2.28	2.84	3.32	3.40	4.29	5.15	5.26	6.38
2	1.02	0.66	3.82	4.03	8.64	8.70	14.37	14.25	19.78	19.44	24.87	24.53	29.79	29.45
3	1.67	0.61	7.81	4.45	19.00	14.36	30.47	26.42	40.43	36.90	48.76	45.67	55.58	52.86
4	5.40	3.19	18.69	13.34	37.89	31.70	52.10	48.51	63.70	62.85	69.99	68.40	75.32	73.79

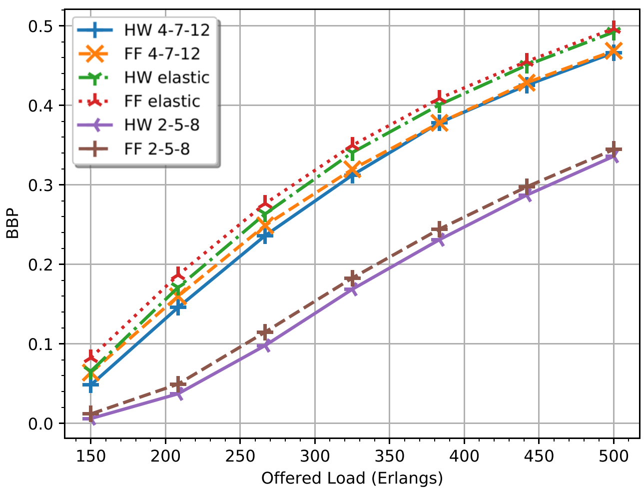

FIGURE 1 BBP for the NSF network topology

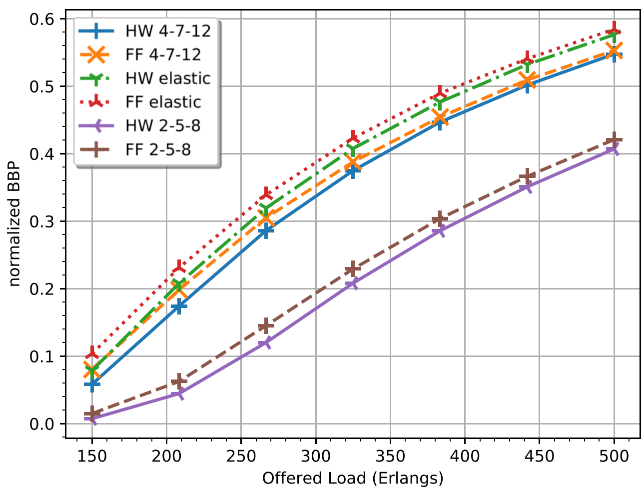

FIGURE 2 Normalized BBP for the NSF network topology

The NSFnet topology is used in Figures 1 to 3. Those two figures compare for the BBP and *normalized_BBP* metrics, respectively, and even though HW's main focus is improving upon *normalized_BBP*, the overall improved network operations translate to BBP improvement. Even though, intuitively, the HW algorithm looks like a trade-off between short and long hop connections that would penalize BBP and benefit *normalized_BBP*, we actually observe comparable improvements for both metrics. This can be explained by the overall more efficient network operations in the case of HW. While normally in FF, a connection will tend to avoid a longer hop connection, thus underusing network resources,

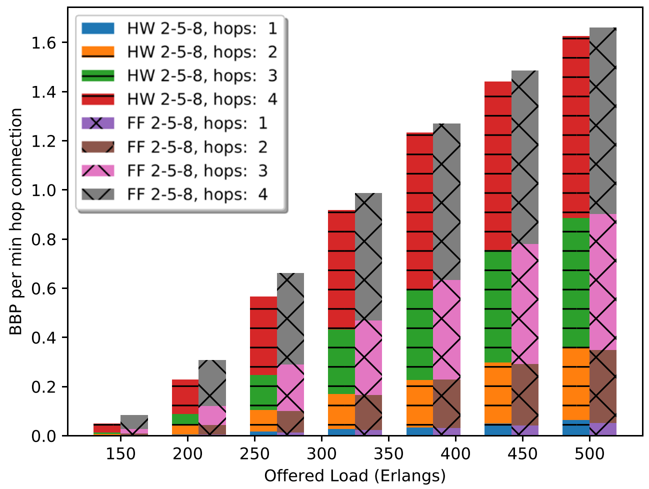

FIGURE 3 Blocking probability per connection hop distance. For each load value, the left bar represents HW and right the FF method. BBP is represented as bar high. Lower sections of the bars represent the blocking probability of low hop-distance connections

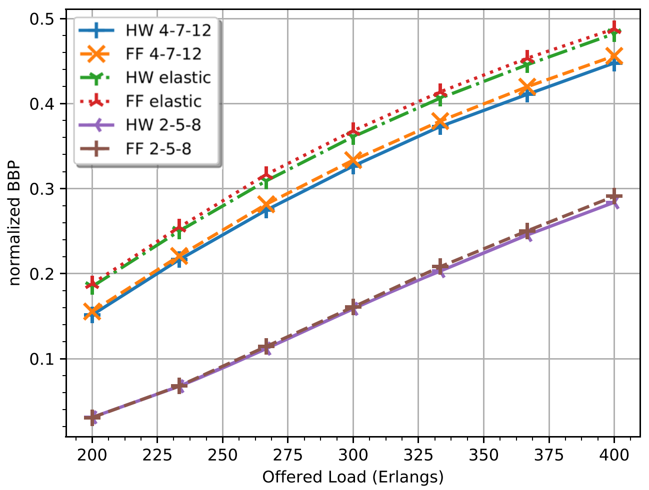

FIGURE 4 Normalized BBP for the metro network topology

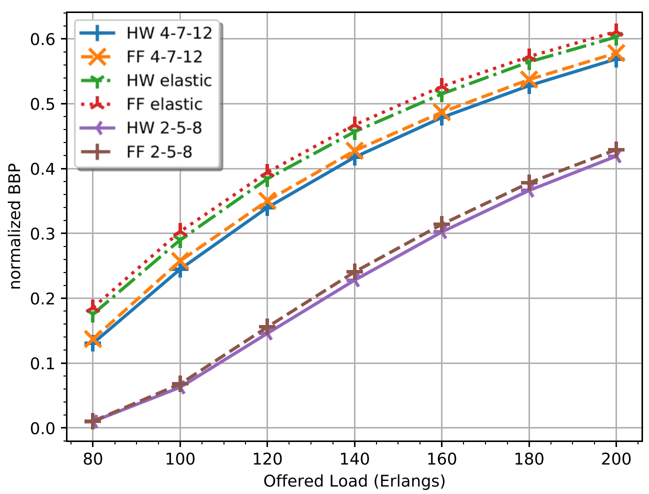

FIGURE 5 Normalized BBP for the ring network topology

in HW, selecting such a path is encouraged. In Figure 3, the BBP per min hop connection are presented in detail. The horizontal-striped left bar represent HW, and the diagonal-stripped right bar represents FF. The difference in BBP performance between low and high hop connections can be observed here. A slight increase in BBP for short connections (one and two hop connections) is translated to a significant decrease in BBP for long connections (three and four hop connections) and improved overall performance.

Similar results can be observed in the case of the mesh network topology (Figure 4) and the 16 nodes ring topology (Figure 5). The three load profiles show different response to when HW is applied; however, a benefit up to 8% is to be expected in the case of *normalized_BBP*. Similar and consistent benefit was also observed in the case of the *BBP* metric (up to 7%).

4 | CONCLUSIONS

In this work, we present HW, a novel RSA algorithm for EONs. It takes the connection's hop distance into consideration, and by limiting short hop connections, long-distance connections are benefited. Our method can decrease bandwidth blocking probability to up to 50% in low offered loads. We show through simulation that HW can be successfully applied to different connection load rules, performs at least as good as FF in all tested network topologies, and a reduction of up to 50% in BBP is seen.

ACKNOWLEDGMENT

This research is cofinanced by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH-CREATE-INNOVATE (project code: T1EDK-05061).

ORCID

Petros Nicopolitidis https://orcid.org/0000-0002-5059-3145
Georgios I. Papadimitriou https://orcid.org/0000-0001-9529-9380

REFERENCES

1. Vizcaino JL, Ye Y, López V, Jiménez F, Duque R, Krummrich P. On the energy efficiency of survivable optical transport networks with flexible-grid. In: European Conference and Exhibition on Optical Communication Optical Society of America; 2012:P5-05.
2. Chatterjee BC, Ba S, Oki E. Fragmentation problems and management approaches in elastic optical networks: a survey. *IEEE Commun Sur Tut*. 2018;20(1):183-210.
3. Chatterjee BC, Sato T, Oki E. Recent research progress on spectrum management approaches in software-defined elastic optical networks. *Opt Switching Netwo*. 2018;30:93-104. Elsevier.
4. Rosa A, Wiatr P, Cavdar C, Carvalho S, Costa J, Wosinska L. Statistical analysis of blocking probability and fragmentation based on Markov modeling of elastic spectrum allocation on fiber link. *Optics Commun*. 2015;354:362-373.
5. Chatterjee BC, Oki E. Dispersion-adaptive first–last fit spectrum allocation scheme for elastic optical networks. *IEEE Commun Let*. 2016;20(4):696-699.
6. Zheng W, Jin Y, Sun W, Guo W, Hu W. On the spectrum-efficiency of bandwidth-variable optical OFDM transport networks. In: Optical Fiber Communication Conference Optical Society of America; 2010:OWR5.
7. Ning J, Shen G. Routing and spectrum allocation in flexi-grid optical networks. In: The 2012 11th International Conference on Optical Communications and Networks (ICOCN) IEEE; 2012:1-4.
8. Comellas J, Junyent G. Improving link spectrum utilization in flexgrid optical networks. *IEEE/OSA Journal of Optical Communications and Networking*. 2015;7(7):618-627.
9. Mavridopoulos SB, Beletsioti GA, Tziroglou GA, et al. A novel hop-distance sensitive approach to elastic optical networks rsa algorithms. In: Proceedings of the 16th International Joint Conference on e-Business and Telecommunications—Volume 1: DCNET, SciTePress; 2019:71-77.
10. Comellas J, Vicario L, Junyent G. Proactive defragmentation in elastic optical networks under dynamic load conditions. *Photonic Network Communications*. 2018;36(1):26-34.
11. Musumeci F, Rottondi C, Nag A, Macaluso I, Zibar D, Ruffini M, Tornatore M. An overview on application of machine learning techniques in optical networks. *IEEE Commun Surv Tut*. 2018;21(2):1383-1408.
12. Mata J, De Miguel I, Duran RJ, Merayo N, Singh SK, Jukan A, Chamania M. Artificial intelligence (AI) methods in optical networks: a comprehensive survey. *Optical switching and networking*. 2018;28:43-57.
13. Shen G, Tucker RS. Energy-minimized design for ip over WDM networks. *J Optical Commun Netw*. 2009;1(1):176-186.

14. Antoniades N, Roudas I, Ellinas G, Amin J. Transport metropolitan optical networking: evolving trends in the architecture design and computer modeling. *J Lightwave Technol*. 2004;22(11):2653-2653.
15. Qiu Y, Fan Z, Chan C-K. Efficient routing and spectrum assignment in elastic optical networks with time scheduled traffic. *Opt Fiber Technol*. 2016;30:116-124.

How to cite this article: Mavridopoulos SB, Beletsioti G, Nicopolitidis P, Papadimitriou GI, Varvarigos E. Hop distance–based bandwidth allocation technique for elastic optical networks. *Int J Commun Syst*. 2020;e4360. https://doi.org/10.1002/dac.4360