

Repositório ISCTE-IUL

Deposited in *Repositório ISCTE-IUL*:

2022-12-05

Deposited version:

Accepted Version

Peer-review status of attached file:

Peer-reviewed

Citation for published item:

Fonseca, P., Cancela, L. & Rebola, J. (2022). Performance analysis of a graph coloring algorithm for wavelength assignment in dynamic optical networks. In 2022 13th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP). (pp. 534-539). Porto: IEEE.

Further information on publisher's website:

10.1109/CSNDSP54353.2022.9907985

Publisher's copyright statement:

This is the peer reviewed version of the following article: Fonseca, P., Cancela, L. & Rebola, J. (2022). Performance analysis of a graph coloring algorithm for wavelength assignment in dynamic optical networks. In 2022 13th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP). (pp. 534-539). Porto: IEEE., which has been published in final form at <https://dx.doi.org/10.1109/CSNDSP54353.2022.9907985>. This article may be used for non-commercial purposes in accordance with the Publisher's Terms and Conditions for self-archiving.

Use policy

Creative Commons CC BY 4.0

The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a link is made to the metadata record in the Repository
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Performance Analysis of a Graph Coloring Algorithm for Wavelength Assignment in Dynamic Optical Networks

Pedro Fonseca², Luís Cancela^{1,2}, João Rebola^{1,2}

¹Optical Communications and Photonics Group, Instituto de Telecomunicações, Lisboa, Portugal

²Department of Science and Information Technology, Iscte - Instituto Universitário de Lisboa, Lisboa, Portugal
paffa@iscte-iul.pt; luis.cancela@iscte-iul.pt; joao.rebola@iscte-iul.pt

Abstract—In the near future (5 to 10 years), dynamic optical networks will be crucial in global optical communications in order to respond to the fast growing of on-demand services. Routing and wavelength assignment (RWA) planning tools must be developed to effectively deal with the dynamic scenarios requirements. In this work, a simulation tool for RWA in dynamic optical networks was developed and wavelength assignment (WA) was implemented, through a recently proposed graph coloring algorithm, named Small-Buckets algorithm, that allows recolorings to occur. Several fiber based networks have been studied and it has been concluded that the Small-Buckets algorithm originates lower blocking probabilities than the ones obtained with the First-fit algorithm. However, to reach this improved performance, the Small-Bucket algorithm requires a larger number of wavelengths and recolorings.

Index Terms—Dynamic Optical Networks, Graph Coloring, Routing and Wavelength Assignment, Small-Buckets algorithm

I. INTRODUCTION

Connections in optical transport networks are usually quasi-static, since they often remain in service for a long period of time [1]. Nevertheless, with the need for rapid response on-demand services, required by applications such as cloud and grid computing, allied with the availability of an increasingly dynamic network infrastructure, it is expected that today's quasi-static optical networks turn into dynamic optical networks in the next coming years [1].

In optical networks, routing and wavelength assignment (RWA) are fundamental network layer functions, which have the target of transporting data in an efficient way. In a dynamic environment, every time a traffic demand arrives, a new wavelength must be found for routing that demand, without changing the wavelengths already in use in the network. The First-fit, Most-used and Random algorithms are examples of heuristic wavelength assignment (WA) algorithms applied to dynamic optical networks [2]. In this dynamic scenario, the blocking probability can be high, especially when several simultaneous traffic demands arrive, and the network is almost at its peak traffic load. In case of network congestion, the network performance can be improved by letting wavelength reconfigurations to occur [3], in order to achieve a blocking probability reduction.

Graph coloring techniques aim at coloring all the graph vertices so that adjacent vertices have different colors [4]. Several algorithms based on graph coloring techniques have already been investigated for solving the WA problem in static optical networks, such as the Greedy, the DSatur (Degree of Saturation) and RLF (Recursive Largest First) algorithms [5], and more recently a Tabu Search algorithm [6]. Nevertheless, to the best of our knowledge, graph coloring techniques for WA in optical dynamic networks have not been studied yet. For this scenario, a simple graph coloring WA algorithm could be based on running the Greedy algorithm every time a new traffic demand arrives, which implies that all the vertices should be recolored since each vertex represents a traffic demand [5]. Another algorithm example is just coloring the demand that has arrived, which would be equivalent to using the First-fit algorithm.

In this work, a graph coloring algorithm, named Small-Buckets algorithm [7], that does not require the recoloring of all the vertices and is, therefore, applicable to dynamic traffic scenarios, has been implemented and studied in the context of WA in optical fiber networks. A comparison with the First-fit algorithm is performed, in terms of the number of wavelengths used, blocking probability and simulation time.

The remainder of the paper is organized as follows. The Small-Buckets algorithm is explained in Section II. In Section III, the developed RWA planning tool is explained and validated with the First-fit algorithm for several network topologies. In Section IV, the results are discussed in terms of blocking probability of a traffic demand, number of colors used, number of recolorings and simulation time. Also, a comparative performance study with the First-fit algorithm is done in this section. Finally, the conclusions are presented in Section V.

II. SMALL-BUCKETS ALGORITHM

In this section, the Small-Buckets algorithm is explained and an example of its application is presented.

In this algorithm [7], a set of buckets is defined, and inside each bucket, several traffic demands can be accommodated (note that each demand is represented by a vertex). Each

bucket has its own set of colors, so that two adjacent vertices do not have the same color. It is assumed a buckets sequence of increasing size, organized in d levels, each level containing s buckets, and a RESET bucket. All buckets at level $i = 0, \dots, d-1$, have capacity for s^i vertices. The RESET bucket is the last system bucket, where all demands are placed whenever all the levels are unable to accommodate a new demand. This bucket has an infinite capacity for vertices (i.e. demands), but like the other buckets, it has a limited number of colors C . The variables used by the algorithm are presented in Table I.

TABLE I
VARIABLES CONSIDERED IN THE SMALL-BUCKETS ALGORITHM.

Variable	Designation	Value
N	Maximum number of vertices to color in the path graph representing the optical network logical topology	—
N_R	Number of vertices during last RESET	—
s	Number of buckets per level	$\lceil N_R^{1/d} \rceil$
d	Number of levels	—
b_{lim}	Maximum number of blocked demands	1000
i	Index of level selected	$\{0, 1, \dots, d-1\}$
n	Index of bucket selected to check availability	$\{0, 1, \dots, s-1\}$
d_i	Level selected to check availability	$\{d_0, d_1, \dots, d_{d-1}\}$
$s_{n,i}$	Bucket n of level i selected to check availability	$\{s_{0,i}, s_{1,i}, \dots, s_{s-1,i}\}$
C	Number of colors per bucket	—

The algorithm starts by placing a new demand in a bucket at level $i = 0$. If this is the last empty bucket at that level, placing it violates the main rule of the algorithm: per level there must be at least one empty bucket. Considering that level $i = 0$ does not have available buckets, all vertices are shifted to the first empty bucket of the next level and then the WA is performed using the specific color set of this new bucket. These vertices shifts are repeated whenever the main rule is not satisfied, up to a limit situation where all vertices have been shifted to and recolored in the RESET bucket. When these level changes occur, vertex recoloring must also occur. In optical networks, a recoloring means that a wavelength reconfiguration needs to be performed, and leads to a reduction of the blocking probability.

Two limit scenarios can be considered. The first one corresponds to perform the maximum number of recolorings i.e., N recolorings per update, where N is the total number of vertices in the path graph. This is possible if, for example, the Greedy algorithm is used for each demand arrival. The second scenario corresponds to perform the minimum number of recolorings, i.e., no recolorings. In this situation, a new color is assigned for each new demand and, so, the total number of colors increases with the number of demands. The Small-Buckets algorithm analyzed in this work is between these two limit scenarios.

The Small-Buckets algorithm will be described next through an example, considering a ring topology with 5 nodes. In this network, it is possible to have a maximum of $N = 20$ distinct paths, considering a full mesh logical topology. The number

of levels considered is $d = 1$. The Small-Buckets algorithm is analyzed considering the first eight traffic demands, that correspond to the first eight steps represented in Fig. 1. The first step of Fig. 1 represents the initial configuration of the algorithm, and the blue colors are used for level $i = 0$ and the green colors are assumed for the RESET bucket. The number of colors in each bucket at level $i = 0$ is $s^0 = 1$.

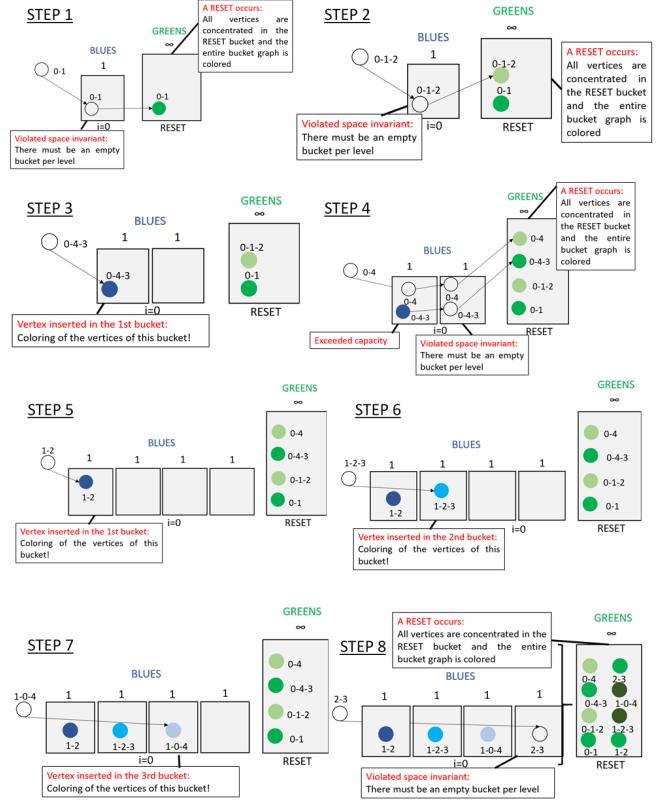


Fig. 1. Small-Buckets algorithm example considering a 5 node ring topology. The first eight traffic demands are represented.

In the traffic demands represented in Fig. 1, there is a violation of the main rule of the algorithm in steps 1, 2, 4 and 8. In the remaining steps, it is possible to place the new vertex in the first empty bucket at level $i = 0$.

In general, if the last bucket of the last level is filled, the system is redefined, by emptying each bucket and placing the demands in the RESET bucket, with a new coloring (steps 1, 2, 4 and 8). This scenario can be observed in step 2, in which the vertices 0-1-2 and 0-1 are colored in the RESET bucket with different green colors. In some steps of the example, as from step 2 to 3, there is a change in the number of buckets per level. When demand #3 arrives, as the RESET bucket is completely filled, the number of buckets in level $i = 0$ is recalculated (following $s = \lceil N_R^{1/d} \rceil$), taking into account the number of vertices in this RESET, $N_R = 2$. Thus, there is the addition of one bucket in level $i = 0$, in the transition from steps 2 to 3. In Fig. 1, the number of buckets at level $i = 0$ is also increased from steps 4 to 5, in this case, for $N_R = 4$.

III. DEVELOPED RWA PLANNING TOOL

In this section, we explain the main building blocks of our planning tool. We also validate the planning tool considering several network scenarios already studied in the literature.

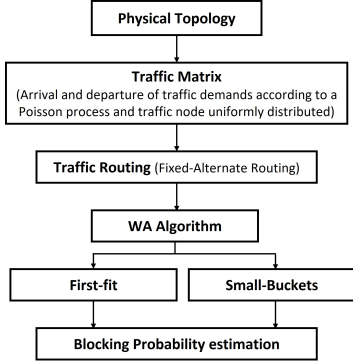


Fig. 2. Developed planning tool diagram.

The planning tool diagram is shown in Fig. 2, with:

- *Physical Topology*: definition of the network physical topology;
- *Traffic Matrix*: in dynamic networks, traffic demands arrive over time. The arrivals and departures of traffic demands are assumed to, follow a Poisson distribution. The total traffic generated by the network is distributed uniformly to each node;
- *Traffic Routing*: to find the optical path between source and destination nodes, the Fixed-Alternate Routing algorithm is applied [8]. This algorithm allows the calculation of two disjoint shortest paths between any nodes pair, before the network goes into operation;
- *WA Algorithm*: after setting the optical path for the traffic demand, a wavelength is assigned to each path. Two WA algorithms are implemented in the simulator: First-fit and Small-Buckets algorithms;
- *Blocking Probability Estimation*: if there is no available wavelength to allocate the demand to the given optical path, the total number of blocked traffic demands is increased. The blocking probability of a traffic demand is obtained by dividing the number of blocked traffic demands by the total number of generated traffic.

The WA algorithms are executed while the number of blocked demands does not reach $b_{lim} = 1000$. If this limit is reached, the blocking probability is calculated and the simulation ends.

The planning tool was validated by obtaining the blocking probability reported in [9] for ring networks, in [10] for the NSFNET and in [11] for the BT network, considering a dynamic traffic scenario and using the fixed-alternated routing algorithm and WA First-fit algorithm.

IV. RESULTS AND DISCUSSION

In this section, the planning tool presented in section III is used to assess the performance of the Small-Buckets WA

algorithm in a 16-node ring and UBN networks and also to compare it with the common First-fit WA algorithm. In particular, the blocking probability, the total number of colors, number of recolorings and simulation time will be assessed.

Considering the Small-Buckets algorithm, for the 16-node ring network, the number of levels is assumed to be $d = \lfloor \ln(240) \rfloor = 5$, while, for the UBN network is $d = \lfloor \ln(552) \rfloor = 6$. As shown in [7], this choice of the number of levels minimizes the number of colors used [7]. When using the First-fit algorithm, the maximum number of wavelengths is assumed equal to the maximum number of colors per bucket.

A. Blocking Probability and Simulation Time

In this subsection, the blocking probability of the 16-node ring and UBN networks is assessed and the algorithms simulation time is estimated. Fig. 3 represents the blocking probability as a function of the average offered load per node considering $C = 48, 56$ and 64 colors and $d = 5$, for the 16-node ring network. The blocking probabilities calculated using the First-fit algorithm are shown for comparison purposes and the computation time of both algorithms is also represented.

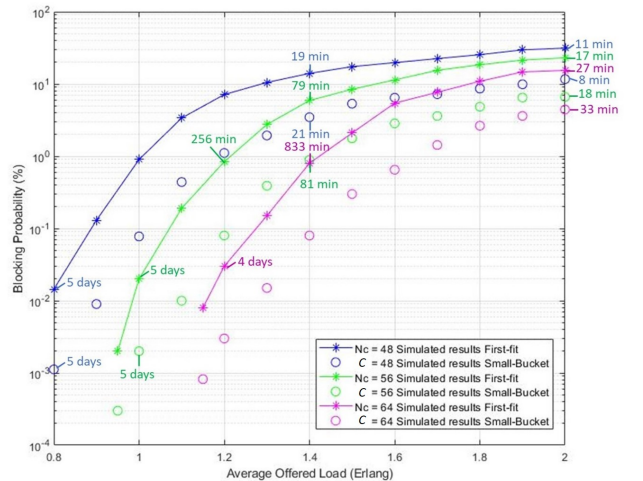


Fig. 3. Blocking probability as a function of the average offered load per node for a bidirectional 16-node ring network with $C = 48, 56$ and 64 colors per bucket considering the Small-Buckets as WA algorithm. The First-fit is also represented for comparison purposes.

As can be seen in Fig. 3, the Small-Buckets with C colors per bucket gives a lower blocking probability, compared to the First-fit algorithm ($N_c = C$ wavelengths), considering the same average offered load. This difference is due to the fact that the Small-Buckets algorithm uses a higher number of colors distributed by s buckets to color the demands, allowing to minimize the blocking probability of a traffic demand. The comparison between the First-fit algorithm assuming N_c wavelengths per optical link and the Small-Buckets considering C colors per bucket and a total of C_{total} colors is not completely fair, because the First-fit algorithm requires less colors than Small-Buckets algorithm.

Fig. 3 represents also some values of the simulation time for different average offered traffic per node values considering the Small-Buckets and First-fit algorithms. Comparing the First-fit algorithm with the Small-Buckets, for the same network traffic, we can conclude that the simulation time obtained is very similar.

Fig. 4 represents the blocking probability as a function of the average offered load per node, for the UBN network, considering $C = 48, 56$ and 64 colors in each bucket, and the Small-Buckets with $d = 6$ and the First-fit algorithms.

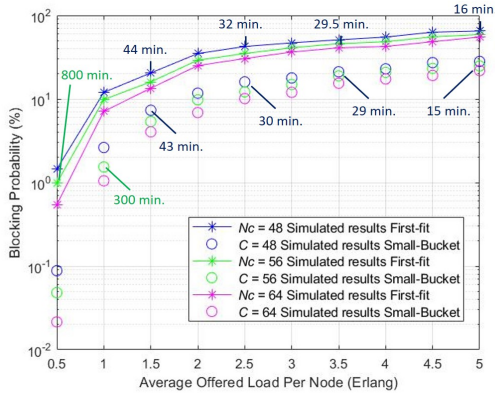


Fig. 4. Blocking probability as a function of the average offered load per node for the UBN network with $C = 48, 56$ and 64 colors per bucket considering the Small-Buckets as WA algorithm.

As can be seen from Fig. 4, once again the Small-Buckets algorithm with C colors per bucket gives a lower blocking probability, compared to the First-fit with $N_c = C$ wavelengths. As stated before, this comparison is not completely fair, because the Small-Buckets algorithm uses a very high number of colors in comparison with the First-fit.

B. Total Number of Colors Used and Recolorings per Update

In this subsection, the number of recolorings and colors is studied. Recolorings consists in the number of wavelengths reassigned in active traffic demands after the arrival of a new demand, which can occur in the RESET bucket and when there are level changes. The number of recolorings as a function of the average offered load per node, for the 16-node ring network with $d = 5$ is shown in Table II. The number of recolorings is obtained by averaging over 1000 blocked demands.

TABLE II
AVERAGE RECOLORINGS PER UPDATE FOR THE 16-RING NETWORK WITH $d = 5$.

A_{node}	$C = 48$	$C = 56$	$C = 64$
1.4 E	2.94	2.73	2.64
1.8 E	2.52	2.57	2.60
2 E	2.77	2.33	2.56

From [6], it can be concluded that the Small-Buckets maintains an adequate coloration of a graph by recoloring a maximum of d vertices per update, each time a demand arrives.

In Table II, the average recolorings per update is around 3, which is smaller than $d = 5$.

Figs. 5 and 6 show the number of recolorings as a function of the number of traffic demands, for $C = 48$ and, respectively, $A_{node} = 0.8$ and 2 E. These offered traffic per node correspond to a total network offered traffic of $A_{network} = 96$ and 240 E, respectively. The average number of recolorings per update is also represented.

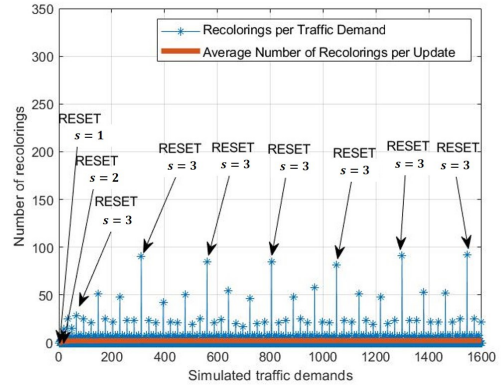


Fig. 5. Simulated number of recolorings as a function of the number of traffic demands, for $A_{node} = 0.8$ E and $C = 48$ considering a 16-node ring.

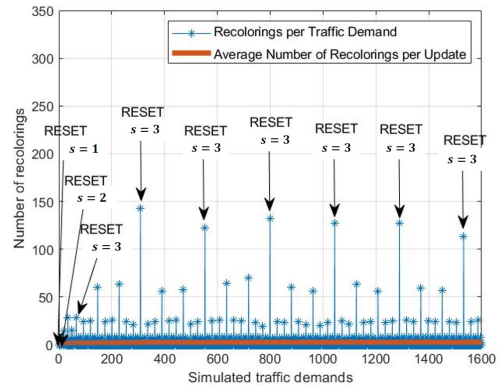


Fig. 6. Simulated number of recolorings as a function of the number of traffic demands, for $A_{node} = 2$ E and $C = 48$ considering a 16-node ring.

For illustrative purposes, only the first 1600 arrivals of traffic demands are shown. In these figures, the RESET situations are highlighted, as well as the evolution of the number of buckets s per level along the simulation, being this number recalculated every time a RESET occurs. From Fig. 5, it can be observed that there are 6 big RESETS, each one corresponding to approximately 90 recolorings. A big RESET is a RESET outside the initial phase of the system. The number of recolorings has several floors, the highest one corresponds to the big RESETS, and the other ones correspond to the recolorings due to vertices level shifts. The number of recolorings due to level changes are approximately 50, 25, 10 and 3, that correspond, respectively, to changes from level 3 to

4, level 2 to 3, level 1 to 2 and level 0 to 1. In the stable phase of the system, the number of buckets per level tends to $s = 3$ and, consequently, the bucket capacity at level i is 3^i vertices. In Fig. 5, the average number of recolorings is approximately 3, after the system initial phase. This low average number of recolorings is reached due to the high number of incoming traffic demands that are placed in the available buckets without causing any recoloring.

In Fig. 6, the number of recolorings corresponding to the floors is larger than the ones found in Fig. 5 due to the higher average load. In the RESET situations outside the initial phase, there are approximately 150 recolorings, whereas in Fig. 5, this number is 90. The vertex coloration rate is much higher than the departure rate of traffic demands, in situations of higher traffic, producing a high bucket occupancy along the simulated traffic demands evolution.

We have also observed, that with $C = 20$, and the same offered load than in Fig. 5 with $C = 48$, the recoloring floor levels are practically of the same magnitude. This is due to the fact that the buckets capacity, i.e. number of vertices, remains the same, despite the decrease in the number of colors per bucket.

For the UBN network, the main difference is the decrease in the number of RESETs in the first 1600 traffic demands. RESETs are less frequent because, in this scenario, the additional level, in comparison with the 16-node ring network with $d = 5$, originates buckets with greater capacity and, thus, reduces the number of RESETs. For example, when the number of buckets per level is $s = 3$, the buckets of the last level ($i = 5$) have $C = 48$ available colors for coloring 243 possible demands ($243 = 3^5$), whereas in the 16-node ring network, the buckets on the last level ($i = 4$) have the same 48 available colors for coloring only 81 possible demands ($81 = 3^4$).

Figs. 7 and 8 shows the total number of colors used, as well as its average value, as a function of the simulated traffic demands for a 16-node ring topology, considering $A_{node} = 0.8$ E and 2 E, respectively, and $C = 48$. The RESET situations, as well as, the number of buckets per level s , are also highlighted. As can be observed in Fig. 7, the number of colors starts to increase from 0 to 70 and a RESET occurs and the total number of colors decreases to 48, at traffic demand #300. This is due to the shift of all demands to the RESET bucket, which has at maximum only $C = 48$ available colors. After traffic demand #300, the number of colors used seems to have a sawtooth behaviour due to the occurrence of big RESETS, which means that the number of colors varies between 48 and 70, every 220 traffic demands. In this situation, the average number of total colors used by the Small-Buckets algorithm tends to a constant value, approximately 50 colors, while the First-fit algorithm uses only an average of 25 colors. This difference is due to the fact that the First-fit algorithm only considers a single coloring space, while the Small-Buckets considers different coloring spaces.

In Fig. 7, the average offered load per node is lower than in Fig. 8. The main difference between these scenarios is the

higher number of colors used in Fig. 8. With the increase of the traffic offered, the number of colors becomes higher, as the buckets are filled more quickly and, as result, more buckets and colors are used.

The Small-Buckets algorithm uses a lower number of colors in the UBN network, in comparison with the 16-node ring network. This is due to the larger capacity of last level buckets ($i = 5$), allowing the reuse of multiple colors. For the UBN network with $d = 6$, the buckets of the last level are able to assign $C = 48$ colors to $3^5 = 243$ vertices. For example, for $A_{node} = 0.35$ E and $C = 48$, the average number of colors used is approximately 40 colors whereas, under the same conditions, for the 16-node ring networks, the average of colors used is approximately 50.

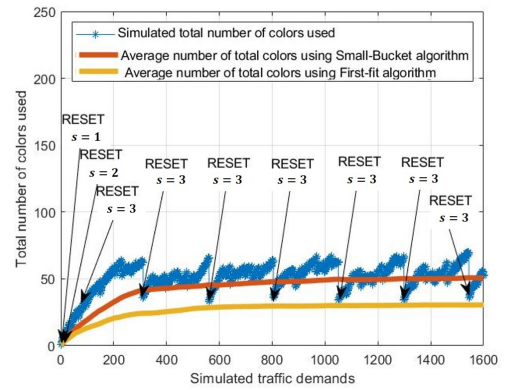


Fig. 7. Simulated number of colors used as a function of the number of traffic demands, for $A_{node} = 0.8$ E and $C = 48$ considering a 16-node ring topology.

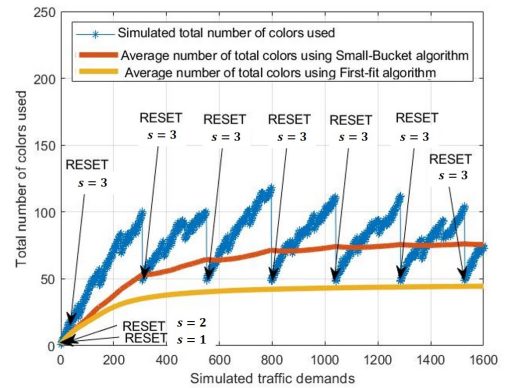


Fig. 8. Simulated number of colors used as a function of the number of traffic demands, for $A_{node} = 2$ E and $C = 48$ considering a 16-node ring topology.

From the results reported in this section, it can be concluded that the Small-Buckets algorithm produces a large number of recolorings, that would require fast tunable transceivers, in order to tune both the transmitter and receiver for the new wavelength, which brings, besides technological issues related

to the tuning speed, cost and management issues, that were not relevant when the WA is performed with the First-fit algorithm [1]. In [12], reconfiguration times of $30 \mu\text{s}$ are reported, which would probably not be enough for our scenario.

In Fig. 3, to achieve a blocking probability of 0.5% with $A_{node} = 1.1 \text{ E}$ similar to the one given by the First-fit algorithm ($N_c = 48$), the Small-Buckets should have $C = 40$. For this scenario, i.e. similar blocking probabilities in both algorithms, the total number of colors used as a function of the simulated traffic demands is represented in Fig. 9.

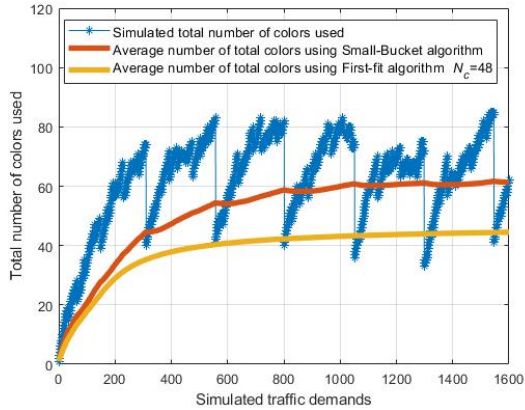


Fig. 9. Simulated number of colors used as a function of the simulated traffic demands and average offered load per node for the 16-node ring network with $d = 5$ and $A_{node} = 1.1 \text{ E}$, when both algorithms give similar blocking probabilities.

From Fig. 9, in order to reach the same blocking probability, the Small-Buckets algorithm requires on average a higher number of colors (approximately 12) compared to the First-fit algorithm.

All results presented in this paper have been obtained for the 16-node ring and the 24-node UBN networks, being the number of levels used in the Small Buckets algorithm, respectively, $d = 5$ and $d = 6$. The choice of the network is directly related to the number of levels considered in the Small Buckets algorithm, $d = \lfloor \ln(N) \rfloor$, with N being the maximum number of optical paths, which depends on the network size. For larger networks, the number of levels would be higher and, consequently, the RESETs occurrences would become less frequent, because the system has a higher number of buckets and with larger capacity. This allows the distribution of demands over more color spaces (buckets), hence, reducing the number of RESET occurrences.

V. CONCLUSIONS

In this work, the Small-Buckets algorithm implementation for WA in optical dynamic networks was described with detail. The Small-Buckets algorithm performance was studied through assessment of the blocking probability, number of colors used, recolorings and simulation time, in two real networks: UBN and 16-node ring network. A comparison with the First-fit algorithm was also performed.

It is concluded that, by considering the maximum number of colors/wavelengths per bucket (C) equal to the maximum number of wavelengths per link (N_c) in the First-fit, the blocking probability obtained using the Small-Buckets algorithm is significantly lower, because the Small-Buckets uses several buckets with capacity C , leading to a much higher number of colors available. In general, a greater number of colors is required when the Small-Buckets is considered as WA algorithm, to obtain similar blocking probabilities as the ones given by the First-fit.

The number of recolorings and colors used can limit the Small-Bucket algorithm application for WA in dynamic networks, since fast tunable transceivers would be required in order to perform the recolorings with higher number of colors. This equipment increase the network cost and management complexity. However, in the Small-Buckets algorithm a traffic demand may not be blocked, because there is a possibility of recoloring, whereas in the First-fit there is no such possibility.

ACKNOWLEDGMENT

This work was supported under the project of Instituto de Telecomunicações UIDB/50008/2020.

REFERENCES

- [1] S. Bigo *et al.*, "Deterministic Dynamic Networks," *Journal of Lightwave Technology*, vol. 37, no. 14, pp. 3465-3474, Jul. 2019.
- [2] J. Simmons, *Optical Network Design and Planning*, 2nd edition, Eds. Holmdel: Springer, 2014, pp. 349-395.
- [3] P. Rajalakshmi, and A. Jhunjhunwala, "Re-Routing at Critical Nodes to Enhance Performance of Wavelength Reassignment in All-Optical WDM Networks Without Wavelength Conversion," *Journal of Lightwave Technology*, vol. 26, no. 17, pp. 3021-3029 Sep. 2008.
- [4] R. M. R. Lewis, *A Guide to Graph Coloring*, Springer, 2016.
- [5] I. Duarte, L. Cancela, and J. Rebola, "Graph Coloring Heuristics for Optical Networks Planning," in *ConfTELE*, Leiria, Portugal, 11-12 Feb. 2021.
- [6] I. Gomes, L. Cancela and J.Rebola, "Exploring the Tabu Search Algorithm as a Graph Coloring Technique for Wavelength Assignment in Optical Networks", in *10th International Conference on Photonics, Optics and Laser Technology*, online, 10-11 Feb., 2022.
- [7] L. Barba *et al.*, "Dynamic Graph Coloring," *Algorithmica*, vol. 81, no. 4, pp. 1319-1341, Jun. 2018.
- [8] R. Ramamurthy and B. Mukherjee, "Fixed-alternate Routing and Wavelength Conversion in Wavelength-Routed Optical Networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, pp. 351-367, Jun. 2002.
- [9] B. Vieira, D. Conceição, J. Pedro, and J. Pires, "Evaluating the Impact of Physical Impairments on the Blocking Performance of an All-optical WDM Ring under Dynamic Traffic," *Journal of Optical Communications*, vol. 28, no. 1, pp. 66-72, Nov. 2007.
- [10] X. Chu, B. Li and I. Chlamtac, "Wavelength Converter Placement Under Different RWA Algorithms in Wavelength-Routed All-Optical Networks," *IEEE Transactions on Communications*, vol. 51, no. 4, pp. 607-617, Apr. 2003.
- [11] D. J. Ives, A. Lord, P. Wright, and S. J. Savory, "Quantifying the Impact of Non-linear Impairments on Blocking Load in Elastic Optical Networks," in *Optical Fiber Communication Conference 2014* ©OSA, San Francisco, 2014, paper W2A.55.
- [12] L. Danh *et al.*, "AgileDCN: An Agile Reconfigurable Optical Data Center Network Architecture," *Journal of Lightwave Technology*, vol. 38, no. 18, pp. 4922-4934, Jun. 2020.