Optical Switching and Networking 22 (2016) 95–104

Contents lists available at ScienceDirect

Optical Switching and Networking

journal homepage: www.elsevier.com/locate/osn

ElasticO++: An Elastic Optical Network Simulation Framework for OMNeT++

Rodrigo Stange Tessinari [a,b,*], Bart Puype [b], Didier Colle [b], Anilton Salles Garcia [a]

[a] *Electrical Engineering Department, Federal University of Espirito Santo, Brazil*
[b] *Internet Based Communication Networks and Services, Ghent University, Belgium*

ARTICLE INFO

Article history:
Received 25 February 2016
Received in revised form
24 June 2016
Accepted 13 July 2016
Available online 15 July 2016

Keywords:
Elastic Optical Networks
RMSA
Fragmentation
Spectrum defragmentation
Algorithms
Simulation
OMNeT++

ABSTRACT

In this work, we present ElasticO++: an Elastic Optical Network Simulation Framework for OMNeT++. A tool created to enable testing in a whole range of routing, modulation, spectrum assignment, defragmentation algorithms, parameters, and topologies. To the best of our knowledge, the proposed framework is the first software available capable of working with fragmentation and defragmentation in dynamic network scenarios. The flexibility available on the proposed tool allows both academia and industry develop new algorithms and techniques for Elastic Optical Networks. The framework provides a set of instruments that allow rapid implementation, testing, and analysis of new algorithms; and enables a common and well-controlled environment for comparing existing algorithms. In its current version, the framework comes with ten traditional already implemented algorithms, which can be used standalone or in combination with new ones. This work describes the architecture and main features, which makes our framework unique. Additionally, we present a case study to demonstrate some of the proposed framework capabilities.

1. Introduction

In the past few years, Elastic Optical Networks (EON) emerged as the "next generation" core network technology [1–3], intended to surpass Wavelength-Division Multiplex (WDM) weaknesses and limitations [4–6]. To keep pace with the always greater demand for bandwidth in core networks, EON relies on Optical Orthogonal Frequency Division Multiplexing (OOFDM) and advanced modulation technologies that enhance spectral efficiency and flexibility [1,6,7]. OOFDM allows the aggregation of multiple sub-carriers to form super-channels, thus changing the paradigm of the network, from fixed-size WDM channels to variable-sized EON channels that can reduce spectrum waste up to 60% [8].

Although several works have pointed EON benefits [4,8], no technology is perfect, and the added efficiency and flexibility comes at the price of increased complexity and new problems, such as spectrum fragmentation [9,10]. When entities with variable sizes start to coexist in the same environment, fragmentation is introduced into the system. Two main strategies can be adopted to address fragmentation problem: (i) prevent fragmentation before it happens (proactive behavior), or (ii) address fragmentation after its manifestation (reactive behavior). Proactive behavior is implemented by specific allocation algorithms [11,12], whereas reactive behavior is accomplished in this context by defragmentation algorithms [13].

A considerable amount of work has been done in both "fronts" (proactive and reactive behavior), introducing a broad range of solutions, which brings us to the following question: "how to compare those solutions and how to identify which one is better suited for the required scenario"? Usually, this is done whenever a new algorithm or technique is proposed, by comparing the proposal to some simple, well-known solution [11,14–16], or by a "survey publication" [1,17–19] that usually brings a qualitative comparison. A major problem with both approaches is the difficulty in comparing published algorithms, particularly because of different simulation scenarios (e.g., network topologies, simulation setup) and, not rarely, missing information regarding simulation/testbed setup or other "hidden assumption and parameter" [1,11,14–18].

In this context, we present ElasticO++: an Elastic Optical Network Simulation Framework for OMNeT++. A framework created to enable testing a whole range of routing, modulation, spectrum assignment, and defragmentation algorithms, parameters, and topologies. We believe that this framework has the potential to become a useful tool to help other researchers in their research projects, especially newcomers. The framework provides

* Corresponding author at: Electrical Engineering Department, Federal University of Espirito Santo, Brazil.
E-mail addresses: stange@inf.ufes.br (R.S. Tessinari), bart.puype@gmail.com (B. Puype), didier.colle@intec.ugent.be (D. Colle), anilton@inf.ufes.br (A.S. Garcia).

http://dx.doi.org/10.1016/j.osn.2016.07.001
1573-4277/

a set of instruments that allow rapid implementation, testing, and analysis of new algorithms; and enables a common and well-controlled environment for comparing existing algorithms. In its current version, the framework comes with ten traditional already implemented algorithms, which can be used standalone or in combination with new ones.

It is important to align expectations at this point. Our intention with this work is to present the capabilities and flexibility of our framework, and not to compare and evaluate algorithm performance; future publications will focus on those comparisons.

This paper is divided as follows: Section 2 presents some EON concepts, whereas ElasticO++ architecture and simulation validation/case study are presented in Sections 3 and 4, respectively. Section 5 concludes this paper.

2. Elastic Optical Networks

This section presents a brief overview about Elastic Optical Networks (EON), focusing at key points necessary to the understanding of this paper. For more extensive information regarding other EON's aspects, refer to the following publications: basics concepts [8], architecture and enabling technologies [4,6], surveys [1,17], and routing and spectrum assignment [18].

2.1. Routing, modulation, and spectrum assignment

One of the key challenges in Elastic Optical Networks is how to optimize resource utilization within the network. This optimization has the potential to decrease network cost and energy consumption as well [4,20]. How to allocate resources between client's requests is defined as the Routing, Modulation, and Spectrum Assignment (RMSA) problem.

At this point, it is important to state different categories of the resource optimization problem. Two are the more common scenarios when discussing resource optimization: *static* and *dynamic* scenarios. Also, it is very common to classify the optimization problem into two additional categories: *online*, and *offline* [21].

In static scenarios all traffic demand is known beforehand; therefore, it is possible to optimize resource utilization according to some metric (e.g., reduce the use of spectrum or smaller hop count average). As all demands are known, the optimization process is done offline and all calculation needed is done beforehand. As computational time is not an issue, optimization algorithms can take several minutes or even days to ensure optimized results. Usually in this static case, fragmentation is not very common, unless the whole scenario possesses multiple phases of allocation/release of resources as demonstrated in [22].

On the other hand, in dynamic scenarios new requests arrive randomly without any previous knowledge of the network. Provisioned connections end randomly as well. Because of the lack of information regarding traffic demand, all resource allocation must be done online, as quickly as possible while clients wait for an answer. Therefore, the processing time of algorithms cannot be too high, and as the RMSA problem is NP-Hard in dynamic scenarios, usually heuristics are used [6,23]. Because of this more unstable environment, it is easily visualized that fragmentation will happen more often than in static scenarios. Fragmentation problem is discussed in Section 2.2.

For each arriving request, an RMSA algorithm needs to evaluate if there are enough resources available to attend that request. If the algorithm can find resources, the request is then accepted by the network, thus creating a new connection and occupying resources. If resources are not available, the request is rejected.

It is important to notice that by resources, we are referring to a spectrum range. Currently, the minimum granularity of spectrum that can be assigned in an EON is 12.5 GHz, and it is known as *frequency slot* or just *slot*. This terminology is used in the rest of the text [24].

The RMSA process starts with a selection of a suitable route between desired source–destination nodes. After routing selection, it is necessary to define how much bandwidth (accounted in slots) is required to transport the requested bitrate. This definition is done by assigning a modulation format, according to link length, signal power, and other physical parameters. An optional spectrum management method can be used. Those management techniques usually consist of splitting the resources into partitions and establishing policies to use it. For example, some policies allocate only demands with same "total path length" [25] or "number of slots" [26] in the same partition. Finally, the last step in the RMSA process consists of finding the necessary number of contiguous slots on the links belonging to the selected route. Fig. 1 (a) illustrates the RMSA process (dashed rectangle represents optional steps). Notice that some restrictions do exist and must be respected to perform the spectrum assign correctly: *spectrum continuity constraint*, which obliges same channels to use same frequency on each fiber link along the end-to-end path; and *spectrum non-overlapping constraint* [24].

To be able to perform the process above a setup phase is needed Fig. 1(b). This setup varies depending on the routing algorithm used. For instance, if the routing algorithm does not perform any offline calculation (i.e., an Online Shortest Path algorithm), the setup process is simpler, consisting only of Topology and Resource detection. Otherwise, if it used a routing algorithm that has an offline phase (e.g., K-Shortest Paths), in addition to the mentioned detection, it is needed to run the routing algorithm and populate a route table with the resulting list of routes.

2.2. Fragmentation/defragmentation

In a dynamic network scenario, incoming requests are established and released in an entirely random fashion. This randomness induces spectral resources to be highly fragmented and consequently, "gaps" are unavoidably introduced leading to the so-called spectrum fragmentation problem. Fragmentation can degrade spectrum utilization efficiency. When connections are uniform, as in WDM networks, fragmentation is caused by wavelength continuity constraint. Though there are wavelengths available on every link that constitute the path, there is not one commonly available on all links. This Inter-link Fragmentation exists in EON as well. However, the focus is on a particular fragmentation problem that occurs when different connections coexist in the same environment. EON's spectrum constraints cause this Intra-link Fragmentation [27].

Any fragmentation leads to inefficient resource utilization and performance degradation, increasing blocking probabilities as the unused slots remain scattered over the links and not enough contiguous spectrum slots may be available for new requests to be established. Several spectrum defragmentation techniques have been developed to prevent performance degradation. The primary aim is to rearrange existing connections in such a way that free slots remain continuous, opening more space for new incoming requests, thus reducing blocking probabilities. Four main spectrum defragmentation techniques are proposed in the literature: (i) Reoptimization technique [28]; (ii) Make-before-break [29]; (iii) Push-and-pull technique [30]; and (iv) Hitless technique [31]/Hop Tuning [13].

In addition to the Defragmentation Algorithm, every time a defragmentation is requested, it is recommended to evaluate if it is indeed required in order to prevent unnecessary operations that, depending on which algorithm is used, may be very costly (e.g., interrupting connections for a period). This prevention can be