Lightness

Low latency and high throughput dynamic network infrastructures
for high performance datacentre interconnects

Small or medium-scale focused research project (STREP)
Co-funded by the European Commission within the Seventh Framework Programme
Project no. 318606
Strategic objective: Future Networks (ICT-2011.1.1)

Start date of project: November 1st, 2012 (36 months duration)

SEVENTH FRAMEWORK PROGRAMME

Deliverable D2.5

Report on simulation results and scalability studies on the proposed DCN network architecture

Due date: 31/10/15

Submission date: 09/11/15

Deliverable leader: BSC

Author list: Jose Carlos Sancho (BSC), Hugo Meyer (BSC), Milica Mrdakovic (BSC), Wang Miao (TUE), and Nicola Calabretta (TUE)

Dissemination Level		
☒	**PU:**	Public
☐	**PP:**	Restricted to other programme participants (including the Commission Services)
☐	**RE:**	Restricted to a group specified by the consortium (including the Commission Services)
☐	**CO:**	Confidential, only for members of the consortium (including the Commission Services)

Abstract

Taking into account that data centers are growing in size, it is important to analyze the scalability of the proposed Architecture-On-Demand (AoD) based optical data center network. Note that it is not only important to calculate how many servers AoD can support in total but also how cost effective is the solution. This report is focused on these aspects of the AoD. It shows its maximum size, investigates in potential scaling bottlenecks, and also proposes techniques to further increase its size. For those purposes, mathematical models have been developed to estimate the size of the AoD. These models are based on the size of the optical devices, OCS, OPS, and hybrid NIC. Simulation results to assess the impact on the performance of the HPC applications have been also carried out. Both single and multi-cluster AoD configurations were considered. The numerical analyses show that the size of the whole AoD network is limited by the size of both the OCS and the output fibers available in the hybrid NICs. The size of the OCS limits the amount of racks in the system whereas the output fibers in the NICS limit the amount of servers per rack. In particular, using a 192-port OCS will result in 190 total racks in an AoD multi-cluster configuration. Different scaling techniques have been proposed to further increase the total amount of servers that the AoD could support. One of the techniques is focused on increasing the number of racks whereas the other technique is focused on increasing the number of servers per rack. The first technique is using smaller OPS in order to allow connecting more racks in a single AoD cluster. Note that this technique is not based on reducing the number of connections to OPS but rather the number of connections that a single application could connect through OPS at the same time. By doing this, the amount of racks in the cluster is significantly increased and thus the amount of running applications that the cluster is supporting at the same time. On the other hand, the second technique is based on increasing the number of servers in a rack by sharing the wavelength among multiple servers. A factor of 8X increase could be achieved. And finally, an economical cost model of the AoD network has been developed. Results show that the proposed network is a cost effective approach to build data center networks compared with the typical electronic-based ones. As it is shown, the full optical network as the one based on AoD reduces substantially the number of optical transceivers required and thus, reduces its costs with no performance penalty.

Table of Contents

Figure Summary

Table Summary

0. Executive Summary

This deliverable focuses on the scalability of the proposed LIGHTNESS DCN based on the AoD. Several studies have been carried out to show the maximum size of the network and its scalability bottlenecks of the network in terms of how many racks and servers the network can support as well as the economic cost. Additionally, there have also been proposed techniques to further scale the amount of servers in this network. These studies can be briefly summarized as follows:

- **AoD network dimension**

The physical dimension of Data Centers based on the AoD-based DCN will be modeled by mathematical equations. The model developed shows the amount of racks and servers that the AoD could support. These studies have been carried out for a single AoD cluster and also for multi-cluster AoD networks. An AoD cluster is composed of one OCS and one OPS. The maximum number of racks that can support an AoD cluster depends on the size of the OCS ports. As expected, larger port count OCS will support OPS with large port count. For example, a 512-port OCS can support 16-port OPS. On the other hand, due to the fiber requirements for OPS a large port OCS does not imply that the number of racks supported is larger. Studies show that the largest number of racks can be found on 320-port OCS, totaling 44 racks. For the AoD multi-cluster configuration the amount of AoD clusters is growing proportionally to the size of the OCS. In particular the total number of racks is 190 when using a commercially available 192-port OCS to interconnect multiple AoD clusters together.

- **Scaling AoD network**

It has been proposed different approaches to scale the amount of servers that an AoD network could support. The first approach is focused on increasing the number of racks by using smaller OPS. The idea is that instead of using a large port OPS it is more convenient for scalability purposes to use multiple smaller port OPS. Using two 6-port OPS the amount of racks could increase to 60 within an AoD cluster.

Another interesting approach to scale the number of servers is to increase the amount of servers per rack rather than increasing the amount of racks as in the previous approach. This can be achieved by sharing the same wavelength among more than one server within a rack. Performance simulation results on HPC applications shows that it could increase the amount of server by 8X without significant impact of the performance of HPC applications.

- **AoD economic cost**

In this last study, it is investigated and compared the cost of the AoD DCN with respect to the equivalent electrical-based DCN. To this purpose, different network architectures have been considered for the electrical-based data center network. A mathematical model to estimate the cost has been developed for each of the different networks considered. Results show that the AoD-based network is significantly cheaper than the electrical network, a 30% reduction in cost is shown. This is due to the reduction on the number of optical transceivers required in the AoD with respect to the electrical ones. The largest cost for AoD is due to the cost of the optical switches and not the optical transceivers as in the electrical networks.

1. Introduction

Data centres are growing in size and complexity to accommodate the ever-increasing demand of more computing resources (i.e. servers) driven by the need to run large number of applications. One of the most challenging issues when scaling out a data centre is the network infrastructure [1] where it must provide high bandwidth and low latency in order to transfer concurrently the traffic from diverse applications in a cost effective way. Optical-based networks have been currently devised as the way to efficiently scale up the bandwidth of current Data centre network infrastructures with lower latency.

In particular, optical devices leveraging on Dense Wavelength Division Multiplexing (DWDM) allows the transmission of several wavelength channels at the same time reaching the barrier of multiple terabit per second per fiber. Recently, a novel type of Architecture-on-Demand (AoD) based hybrid optical switched Data Centre network architecture has been proposed in LIGHTNESS combining both Optical Circuit Switching (OCS) and Optical Packet Switching (OPS) [2]. This new network architecture is able to quickly forward flows to either OCS or OPS depending of the characteristic of the flows. For example, large and long live flows could be forwarded to OCS and short live flows to OPS taking advantage of its statistical multiplexing characteristic. An AoD cluster will contain a large OCS and also OPS devices to handle different kinds of traffic within each cluster. Multiple AoD clusters could be connected through an OCS and/or OPS to scale up modularly offered by this network architecture to build a multi-custer data center.

It is important to foresee the maximum dimension of this AoD network architecture proposed in LIGHTNESS in order to accommodate the large number of servers in current data centers. To investigate this, it has been following three approaches. The first one, it has been developed mathematical equations to model the dimension of both one AoD cluster and multi-cluster configurations. These models will provide valuable insights on the scalability limits of this network architecture and at the same time to quantify the dimension of the network. The second approach, it has been provided simulation results of this optical network with real traces coming from High Performance Computing (HPC) applications to evaluate the impact of the network in these applications. A comparison with its counterpart electrical network architecture has been also provided, namely InfiniBand network. These simulation results will provide important insights on the performance scalability of the network. The last approach, it has been provided an important study on the scalability of the network in terms of cost. Results indicate that the LIGTHNESS network can outperform the electronic switch based network in terms of costs.

Furthermore, we have investigated different approaches to scale up further the dimension of this AoD network architecture. These approaches have been analysed with both mathematical models and evaluated their impact on performance of HPC applications through accurate simulations of the network.

2. AoD cluster overview

The AoD based optical data centre network (DCN) is a flat network architecture where a diverse set of multiple passive and active optical devices are plugged into an optical backplane to provide dynamic, programmable, and highly available DCN connectivity services while meeting the requirements of new and emerging DC (data centre) and cloud applications. LIGHTNESS data plane integrates innovative optical switching technologies including programmable hybrid optical Network Interface Card (NIC), optical Top of the Rack (ToR) switch, optical packet switching (OPS) and optical circuit switching (OCS), controlled and operated by a Software Defined Networking (SDN) based control plane for enhanced programmability of different network functions and protocols. Using the power of optics enables DCs to effectively cope with the high-performance applications' demands.

The architecture-on-demand (AoD) interconnects all the switching elements and enables the flexible configuration of the intra- and inter-cluster communication for the flat DCN. Figure 1 shows an AoD based cluster. NIC aggregates the traffic from the dedicated server and transmits the data to the all-optical ToR for intra- and inter-rack communication. OPS and OCS backplane interconnects all the ToRs and are specified to handle inter-rack traffic.

The interface between the control plane and the data plane is responsible for enabling the SDN-based control and operation of the DCN. This interface is implemented using extended OpenFlow (OF) protocol, thus some particular features of proposed AoD architecture can be supported. On top of each network device (i.e. OCS, OPS, ToR and NIC) there is an Agent which translates messages coming from the SDN-controller to an actual configuration of the device. On the other hand, the Agents collect monitoring information from the devices and send it to the SDN controller.

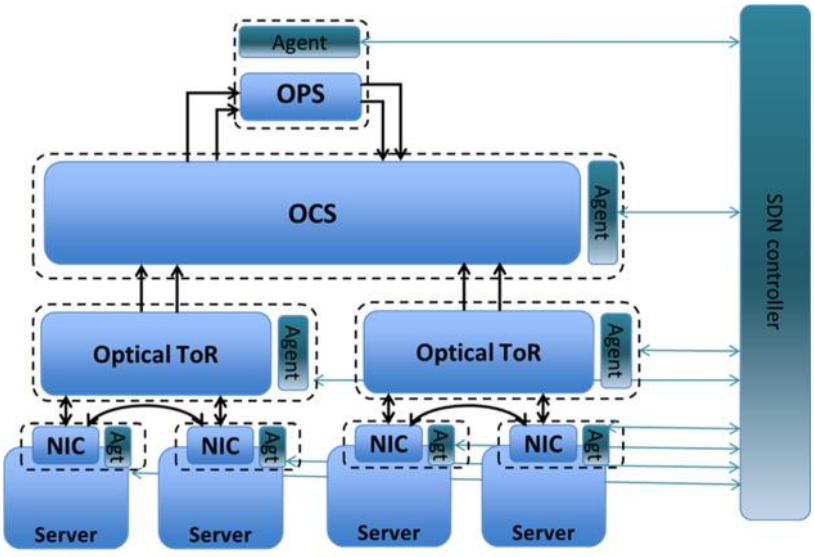


Figure 1. AoD cluster

2.1. Optical devices

This section provides a brief overview of the key optical devices employed in the AoD network proposed in LIGHTNESS. These devices are the network interface cards that deal with the optical/electrical conversions and provide optical connectivity among the servers in a rack; the top-of-the-rack that provides the interface of the rack to others racks in the system; and finally, the last optical devices are the optical switches based on OCS or OPS technology that connect different racks in the system.

2.1.1. Network interface card

Each server contains a network interface card (NIC) which does the conversion from electrical to optical domain (E/O) and vice versa. There is a limited size buffer to store temporally packets that are received directly from the server before the E/O is performed. Packets that are delivered to the optical network are not yet discarded by the NIC until an ACK notification is received from the optical network indicating that the packet was delivered properly. This approach guarantees that no packets will be lost. This is important because the OPS could drop packets due to packet contentions (as it will be shown in sections below). Retransmissions of dropped packets are performed automatically by the NICs. Each NIC has a connection with its server and two types of interfaces. Connections to each server in the rack are 10Gb/s fiber links, while interfaces to ToR can provide ten times higher bandwidth from each server.

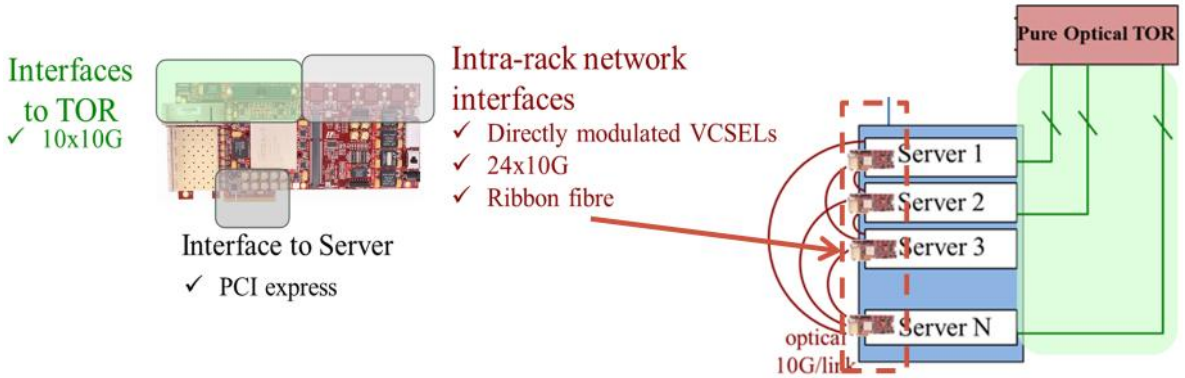


Figure 2. Server's NIC and server's connections within a rack

2.1.2. Top of the rack

Top-of-the-rack (ToR) switch reside on the top the rack and provides the interface to the OCS in the AoD network. Figure 2 shows the proposed optical ToR and its connections with the servers in rack. Each server

has a direct connection to the ToR using one wavelength per server. Each NIC provides a number of fibers to connect to other servers in a full mesh topology to every other server using direct fibers. In the actual NIC implementation 24 fibers are provided to connect to other servers within a rack. The number of fibers in the NIC could limit the number of servers that a rack can support because there is a need to connect every server with other server within the rack. A NIC that provides a large number of fibers then a large number of servers could be allocated in a rack. The ToR makes the aggregation of multiple wavelengths within a fiber by using a common optical multiplexer as shown in Figure 3. A demultiplexer is also used in the other direction to separate each wavelength in the fiber. State-of-the-art Dense Wavelength Division Multiplexing (DWDM) could support theoretically up to 80 separate wavelengths or channels of data can be multiplexed into a single optical fiber. Typically, practical DWDM supports only 80 or less number of wavelengths. This could pose another limitation of the number of servers that a rack could be supported under the AoD architecture. In summary, the maximum number of servers per rack in the AoD network can be modelled by the following expression:

$$N \quad = m \quad (N \quad , N \quad ht:)$$

Where NIC_{ports} is the number of ports that the NIC support, and $N_{wavelenghts}$ is the number of wavelengths supported in the fiber in the actual AoD network.

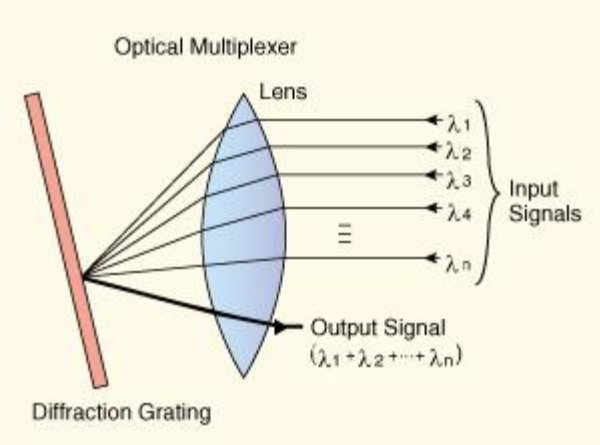


Figure 3. Optical multiplexer [3]

2.1.3. Optical circuit switching

Optical Circuit Switching (OCS) is an optical networking technology where the device is configured to establish a circuit, from an ingress optical port to an egress optical port, by adjusting internal optical connections that connect the ingress to the egress. All the light coming from the ingress is traveling with no delay in an all-optical manner to the configured egress port. This device requires setting up the path from the ingress to the egress port, prior to the transmission and this time interval could be high. Currently, OCS switches have a

setup time of 25ms (mirrors configuration). In addition, OCS commercial devices up to 192 ports can be found such as the OCS Polatis Series 6000n [3], but larger port up to 320 is feasible.

2.1.4. Optical packet switching

The architecture of the scalable Optical packet switching (OPS) is shown in detail in the Figure 4. An OPS is composed of *F* modules and each of them manages the *n* wavelengths coming from an input fiber. On the other hand, there are *F x F* fibers in the output, each of the *F* output ports has *F* fibers that are connecting to the ToR as it was described above. All the *F* modules are working in parallel in order to minimize processing time and thus the switching latency [4]. Note that the switching latency is port-count independent unlike in electronic switches, and thus it can provide a lower processing time.

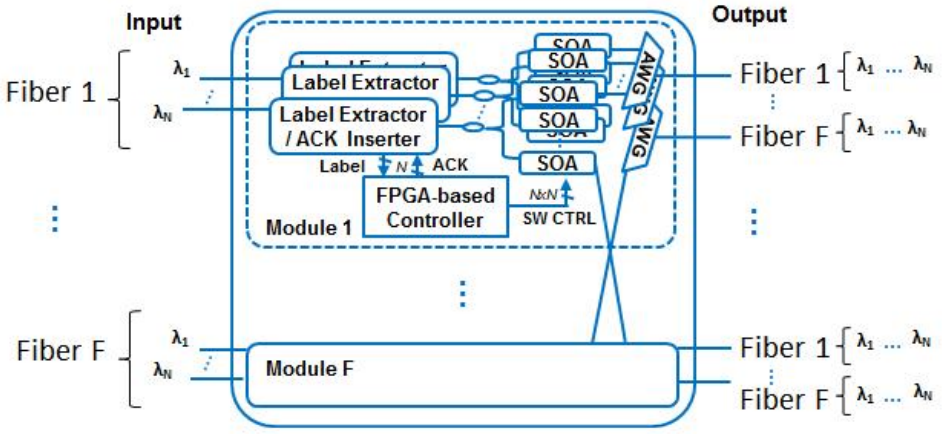


Figure 4: Scalable OPS

Each module consists of several Label extractors and ACK inserters, several semiconductor optical amplifiers (SOAs), several arrayed waveguide gratings (AWG)s, and a FPGA-based switch controller.

A label extractor separates the optical label from the optical packet by using a Fiber Bragg Grating (FBG). The packet label provides the packet destination. It is converted from optical to electrical domain in order to be processed by the FPGA-based switch controller. The FPGA-based switch controller will control the SOAs to forward the packets to certain destinations. The optical payload is then fed into the SOA based broadcasting and selecting stage. Finally, the AWG multiplexes the wavelengths into the selected fiber.

By checking the destination information carried by the labels, the FPGA-based switch controller detects and resolves the possible contentions. Collisions can only occur when packets coming from the same input fiber are destined to the same output fiber, in which case the packets will overlap in time domain resulting in errors at the receiver side. In case of a collision, only one packet can be forwarded and the rest are dropped. The controller generates a positive acknowledgment (ACK) informing the corresponding NIC about the packet that got forwarded whereas one or more negative ACKs (NACK) are generated to inform of the corresponding packets dropped and have to be re-transmitted. The ACK/NACK is sent back to the NIC within the same optical

link [5]. The base-band ACK/NACK signal is easily extracted at the NIC by using a 50 MHz low pass filter, to remove the label information at RF frequencies.

OPS are not symmetric in the number of input and output ports. The number of input ports coming from the NIC to the OPS is *F* (each with *n* wavelength channels), but the number of output ports from the OPS to NIC is *F* x (*F* - 1). This approach is applied in order to avoid collisions in the OPS when multiple optical packets coming from different NICs want to go to the same destination NIC. The number of output fiber can be reduced to F if fix wavelength converter are employed in architecture to avoid collisions. Each NIC will transmit through a different output fiber. For example, a packet from NIC *i* that wants to transmit to NIC *j* will use the output fiber *i* in the NIC *j*. Typically, the number of inputfibers is equal to the number of NICs, *F* = *R*, and thus the number of output fibers is *R* x *R* regardless of the number of wavelengths per fiber.

2.2. AoD cluster dimension model

In this section, an analysis about the number of racks that an AoD cluster can accommodate is presented. As it was described before, an AoD cluster is commonly composed of one OCS, one OPS switches on top of the OCS, and several racks that are connected to the OCS. For this study, it is calculated the maximum size of one OPS that an AoD cluster supports taking into account the number of ports available in the OCS. The number of servers is proportional to the number of racks and it can be directly calculated by multiplying the number of racks by the number of servers per rack supported. We are assuming that racks connect to the OCS through one fiber, thus one rack uses only one OCS port. This fiber will provide connectivity to all the servers in the rack. It is assumed that a different wavelength is provided for each server in the rack.

An OPS requires two different set of ports, input and output ports. As it was described above, the number of OPS input (OPS_{input}) and OPS output (OPS_{output}) ports are different. The relation between input and output ports is given by, $OPS_{ouput} = OPS_{input}$ x (OPS_{input} -1). Therefore, the number of OCS ports required to connect one OPS is given by Eq. 1.

$$O_p = O_{in} + O_o = O_{in} + O_{in} x(O_{in} - 1) = O_{in}^2 \quad (1)$$

The number of ports required by OPS is proportional to the number of input ports, but it is not growing linearly. For this reason, only some specific number of input ports is supported for each particular OCS size. The OPS maximum size supported (N_{OPS}) for a given OCS size is calculated taking into account that the OPS input ports will connect each rack in the AoD cluster and each rack will consume twice the number of ports in the OCS because it will need these two connections: from racks to OCS and from OCS to the OPS. Therefore, the maximum number of OPS input ports, and thus the maximum number of racks that we can connect to the OPS will be given by,

$$N_o = 2 \text{ x } O_p = 2 * O_{in}^2 \quad (2)$$

And hence, the maximum OPS size will be given by Eq. 3 (defining OPS_{input} as N_{OPS}).

$$N_O = \sqrt{\frac{N_o}{2}} \quad (3)$$

This formula calculates the number of racks that can connect to the OPS based on the size of the OCS. In particular, for a 1923-port OCS the maximum number of OPS input ports will be nine (N_{OPS} = 9).

Note that the number of racks could not fully use all the available ports that an OCS provides because the total number of supported OPS ports is not linear. However, the available remaining ports in the OCS could still be used to connect racks. These additional racks could only connect to other racks through uniquely using the OCS. The total number of racks is given by,

$$R_{t} = R_O + R_{OC} \quad (4)$$

where R_{OPS} is number of racks which will support the communication through the OPS and R_{OCS} is the number of racks which will use uniquely OCS. Note that R_{OPS} could also use OCS for communication if it is more efficient than the OPS. As stated above, the number of racks supported for the OPS will be the size of the OPS ($R_{OPS} = N_{OPS}$). On the other hand, R_{OCS} is derived from the unused OCS ports. Number of OCS racks is given by,

$$R_O = N_O - N_O = N_O - \sqrt{\frac{N_O}{2}} \quad (5)$$

For example, for a 192-port OCS, the resulting ROPS = 9 and ROCS = 30. These 30 ports can be used for connecting racks through the OCS. For illustration purposes, Figure 5 shows full utilization of OCS ports when applying this architecture. Devices that transfer data through the OPS are shown with green, while the devices that communicate through the OCS are shown with blue colour. Fibers that connect racks which will use only OCS are multi-mode, bidirectional fibers.

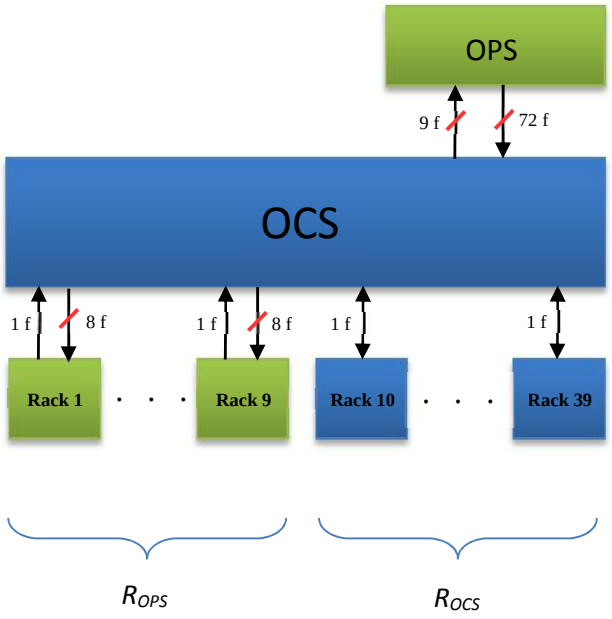


Figure 5. Distribution of racks in an AoD cluster

With the increasing size of the OCS, the potential size of OPS increases as well. Table 1 shows leading vendors in optical switching with their OCS products and the prototype which is a tendency in optical switching. These switches are used in Figure 6 in order to analyse the maximum number of input ports of OPS according to the OCS size.

OCS switch	Vendor - model
32x32	Polatis - Series 1000
48x48	Polatis – Series 6000 Lite
160x160	Calient – S160
192x192	Polatis – Series 6000
320x320	Calient – S320
512x512	Prototype

Table 1. OCS leading optical vendors' models

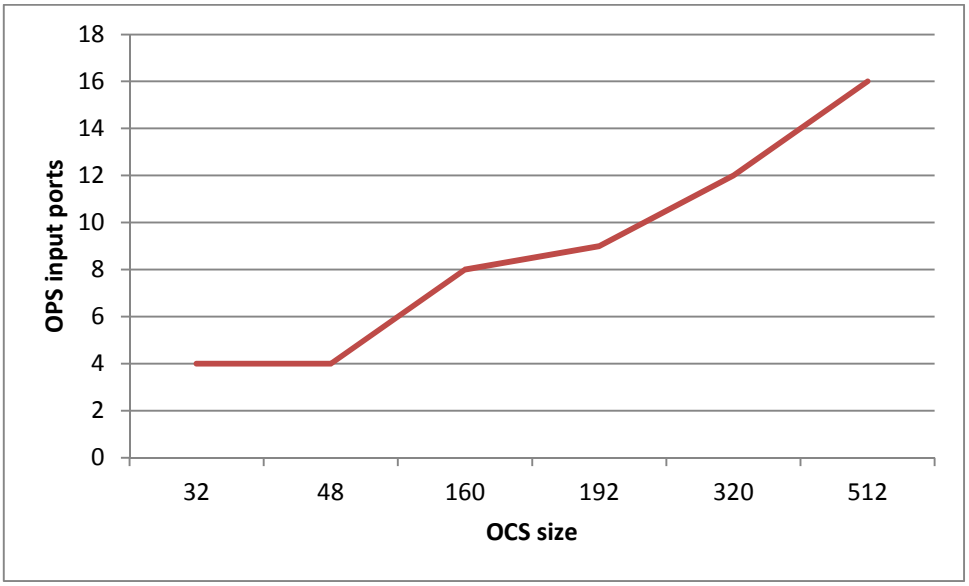


Figure 6. Increasing OCS increases size of OPS

The relation between number of racks and OCS size is shown in Figure 7. It could be seen that R_{OPS} grows as OCS size grows, but it is not the case with R_{OCS}. R_{OCS} is fully depended on the remaining ports in OCS, the ports which are not used for OPS. Each remaining port can be used as a connection for one OCS rack, because the rack that will use only OCS needs just one bidirectional, multi-mode fiber. Note that the largest number of racks is found in 320-port OCS and not in 512-port OCS. Figure 8 shows the number of servers while increasing the OCS size. Note that we are assuming 40 servers per racks to perform this estimation.

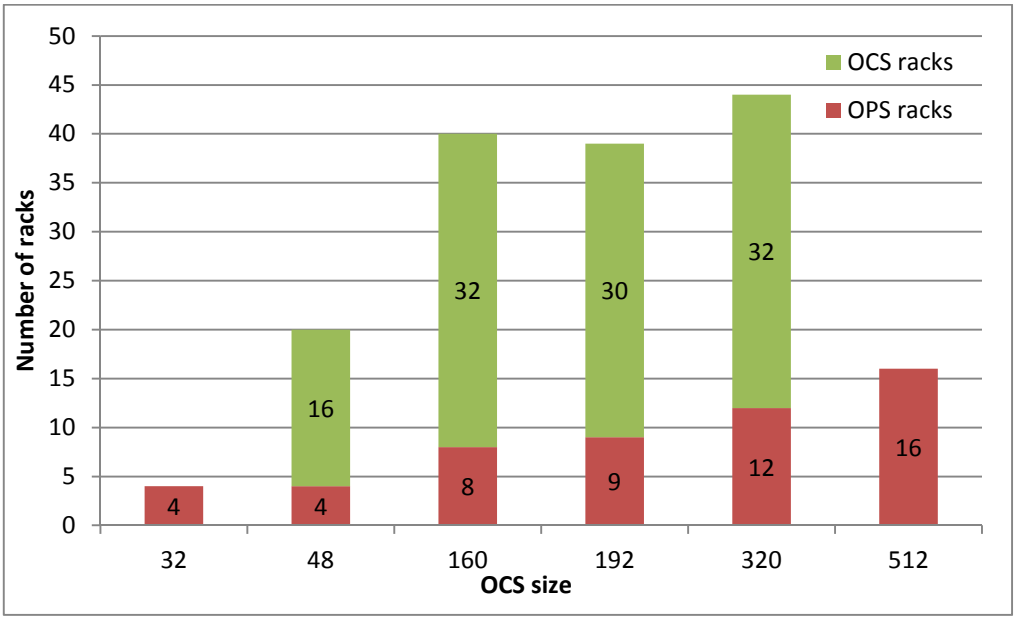


Figure 7. Number of racks – changing OCS size

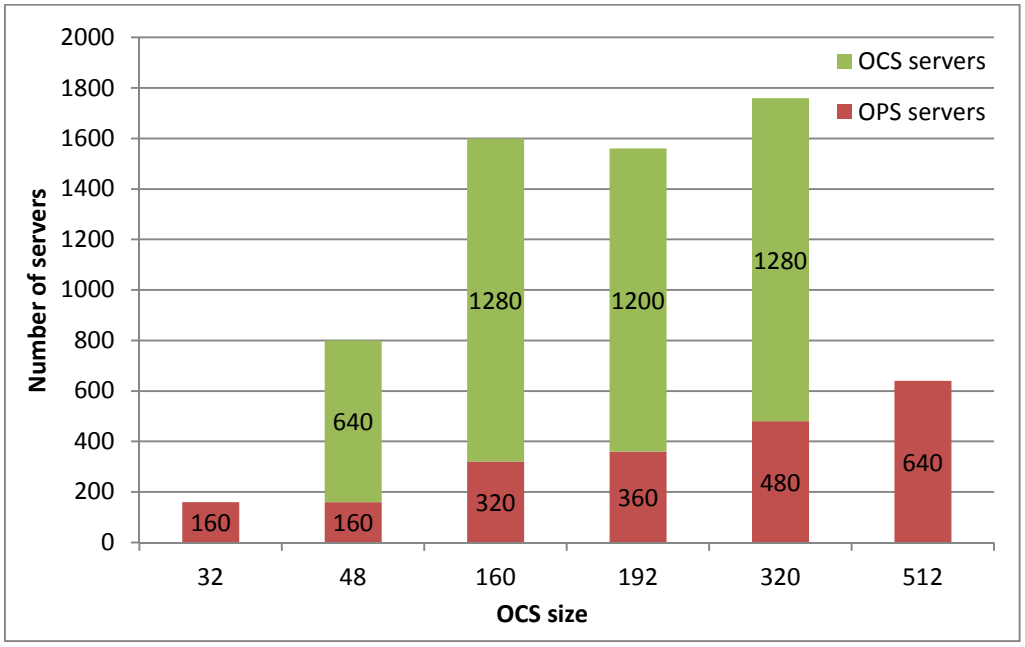


Figure 8. Number of servers – changing OCS size

2.3. AoD multi-cluster configuration

The same AoD cluster concepts introduced in the previous section are used to model the multi-cluster AoD configuration. The multi-cluster AoD architecture design is extended to the inter-cluster DCN architecture, as shown in

Figure 9 (explained in detail in Deliverable D3.1). With all clusters using the same AoD network, another optical transport network (created using OCS and OPS) is used as interconnection between clusters. All clusters and inter-cluster OPS switches are connected to the inter-cluster OCS. ToRs in different clusters can communicate with each other through relayed OCS links or OPS modules provided by inter- and intra-cluster optical transport network.

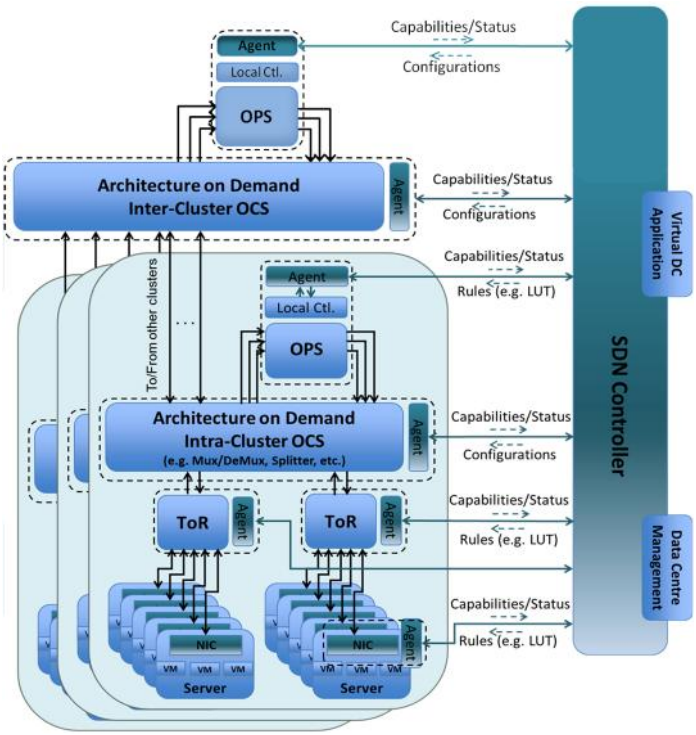


Figure 9. Multi-cluster approach

The main advantage of the LIGHTNESS DCN design is that interconnections between ToRs in the same cluster can be reconfigured according to the traffic needs. The traffic capability between each ToR pair could be dynamically adapted and expanded to the maximum of 100% utilization of connections provided by each ToR.

Any link from any ToR can be assigned to any switching module on demand, so that programmable intra-cluster DCN can be constructed dynamically in terms of function as well as topology. Like that, heavy traffic loads from the same ToR can be split and transferred through different network paths. Traffic loads from overwhelmed ToRs can be isolated by rearranging topological locations of these ToRs in the DCN. Heavy traffic loads can be placed to the branches with fewer loads. Like this, the traffic in each branch is balanced. Such programmability works as for intra also for the inter-cluster DCN, thus the overall network utilization and network performance are improved.

Redundancy is an advantage that worth mentioning since faulty modules in one cluster can be replaced by modules in the whole DC, without human interfering. Network topology and operation can be decided

according to the QoS required by each task, and also the network resources available or assigned to each cluster in the DC.

2.4. AoD multi-cluster dimension model

Several clusters can be connected through a similar optical network to the one used inside the cluster. In this section we first analyse the inter-connection with one inter-OCS. This approach requires fibers and OCS ports that will connect to inter-OCS in order to reach other AoD clusters. This multi-cluster approach is analysed below.

Taking into account inter-cluster connections, the intra-OCS needs several types of connections that are modelled as following,

$$N_O = 2 * T_O + T_O + N_{i}$$ (6)

Given that $2*T_{OPS}$ represents total number of connections for racks that will transfer data through the OPS, T_{OCS} represents connections that will use OCS ports for their data transfer, and N_{inter} is the number of inter-connections between different clusters. N_{inter} could be calculated as the sum of the connections needed to connect OPS racks and also the OCS racks,

$$N_{i} = N_{i} + N_{i}$$ (7)

Then, the number of intra-OCS ports is represented by,

$$N_O = 2 * T_O + T_O + N_{i} + N_{i}$$ (8)

N_O shows the number of ports needed for connecting one OPS switch. Furthermore, the intra-OCS needs this number of ports towards the OPS switch, but also towards the racks that will use OPS. That is the reason for the expression $2*T_{OPS}$ where T_{OPS} is given by,

$$T_O = R_O + R_O * (R_O - 1) = R_O^2$$ (9)

R_{OPS} is the number of racks that will use OPS and at the same time the number of OPS input ports. Equation 3 shows the connection between the number of intra-OCS ports and maximum possible input ports of OPS switch. OPS input ports are replaced with R_{OPS} ($N_{OPS} = R_{OPS}$) and the inter-connections for all OPS racks are also considered. The maximum number of OPS racks is counted as:

$$R_O = \sqrt{\frac{N_O - N_{i}}{2}}$$ (10)

Value of $N_{interOPS}$ is located inside the range $1 \leq N_{interOPS} \leq R_{OPS}$. Here we will consider that each rack has its own fiber for data transfer outside the cluster, i.e. $N_{interOPS} = R_{OPS}$. By placing this expression in Equation 10, we can obtain the variable R_{OPS} using a quadratic expression. Intra-OCS has limited number of ports represented with N_{OCS}. With $N_{OCS} = 192$, then the solution of the quadratic equation is given by,

$$R_O = \frac{-1 \pm \sqrt{1^2 + 4*2*1}}{2*2} = 9.55$$ (11)

The integer part of the solution is taken into account because the number of switches cannot be expressed with decimal values, so the value of R_{OPS} will be 9. The decimal part shows that some ports will remain unused by OPS. Once the number of ports that will be consumed by the OPS is known, we can count the ports that remain for available uniquely to connect the OCS,

$$T_O + N_{ii} = N_o - 2 * T_O - N_{ii} \quad (12)$$

T_{OCS} is equal to the number of servers that will use OCS ($T_{OCS} = R_{OCS}$), because each rack consumes just one intra-OCS port for the data transfer through the OCS. It can be considered that each rack has its own inter-connection towards the inter-OCS, which means that $N_{interOCS} = R_{OCS}$. By inserting these values into Equation 12 it is obtained the maximum number of racks that can use just the OCS for their data transfer,

$$R_O = \frac{N_o - 2*T_O - N_{ii}}{2} = 10.5 \quad (13)$$

The integer part of the result provides the values $R_{OCS} = N_{interOCS} = 10$. Depending on the size of inter-OCS (I_{OCS}), the maximum number of clusters in a multi-cluster environment is given by,

$$C = \frac{I_O}{N_{ii} + N_{ii}} \quad (14)$$

Figure 10 shows one cluster with all mentioned connections, it shows just one part of the multi-cluster architecture with one cluster with all necessary connections. Green colour indicates racks that make use of the OPS, while blue refers to racks that use uniquely the OCS. As it can be seen, traffic from all racks goes through the inter-OCS and the racks use either OPS or OCS inside their cluster.

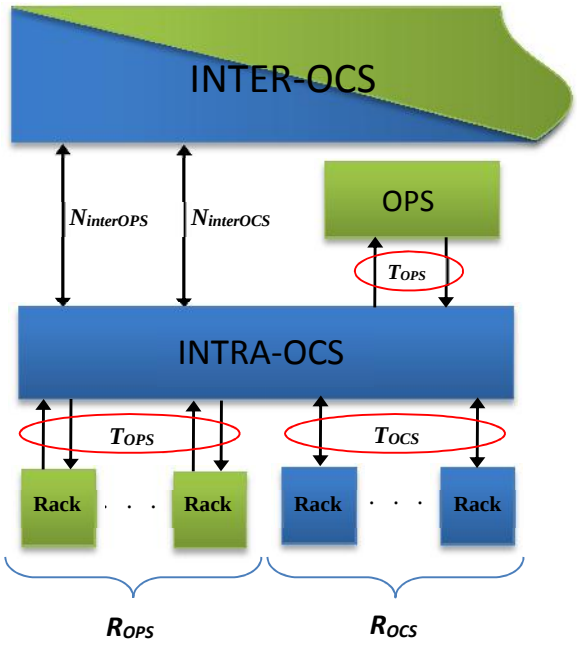


Figure 10. Inter Cluster Connections using one Inter-OCS

Figure 11 shows the full architecture with specific number of connections. We consider that each intra-OCS switch has 192 ports, while inter-OCS switch has 64 ports. When intra-OCS switch has 192 ports, the maximum number of OPS input ports inside the cluster are 9 (Eq. 11). There are 21 ports left for OCS racks and their connections (Eq. 13), more precisely for their intra- and inter-connections. Like this, there are 10 ports for connections with OCS racks and other 10 for connections with inter-OCS.

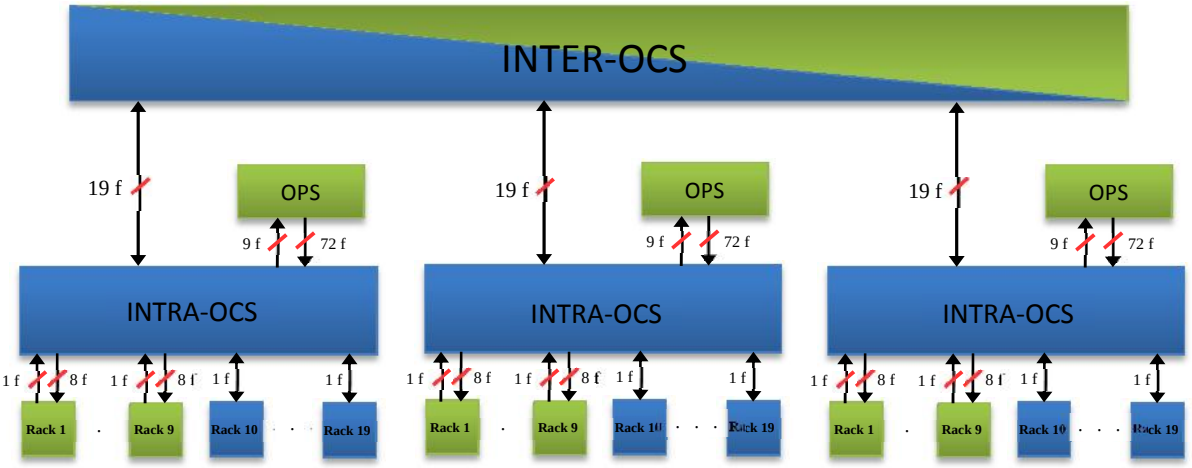


Figure 11. Multi-cluster architecture

The maximum number of connected clusters is calculated by the Equation 14. Figure 12 presents the maximum number of clusters that can be connected using one inter-OCS without using any OPS on top of the inter-OCS. Each cluster is the same size as in

Figure **11**. Values on X-axis represent IOCS which varies from 32 to 512 ports. Note that if each cluster is connected with only one fiber between intra and inter-OCS, the maximum number of clusters will be equal to the size of inter-OCS switch, i.e. to its number of ports.

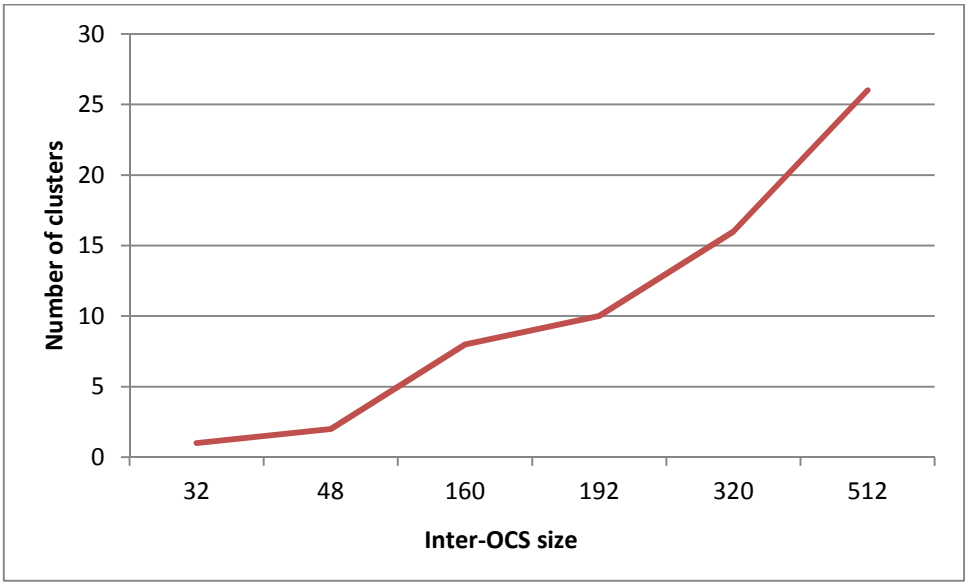


Figure 12. Number of cluster in a multi- cluster AoD configuration

Taking into account that the usage of inter-OPS requires a lot of ports and reduces significantly the number of inter-connected clusters and servers, it has not been included in the analysis. Then, it is not advisable using Inter-OPSs switches if there is a need for high scalability with high number of clusters.

Figure 13 summaries the amount of clusters, OPS racks, and OCS racks that could be connected using uniquely one 192-port inter-OCS. The number of OPS and OCS racks are obtained applying the maximum possible number of OPS input ports and the rest of the ports are used for OCS connections.

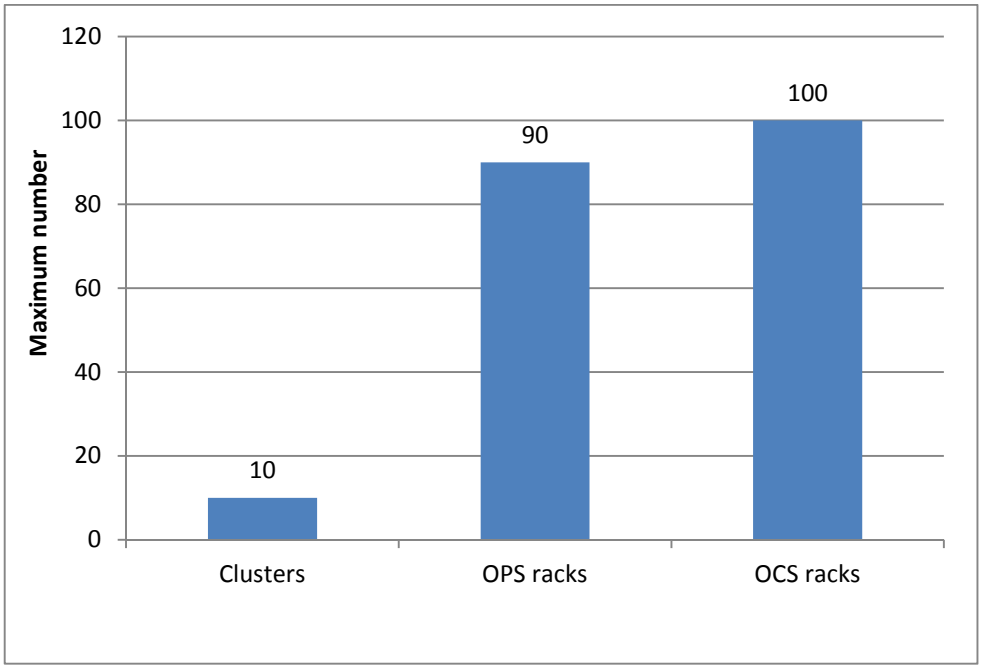


Figure 13. Multi-cluster AoD dimension when using a 192-port inter-OCS

2.5. AoD simulation framework

The simulator developed for the LIGHTNESS project is called *DimlightSim* and it has been developed using Omnest [6]. *DimlightSim* models the main components of the network architecture proposed within the LIGHTNESS project, namely Network Interface Card, (NIC), Top-of-Rack (ToR) Switch, Optical Packet Switch (OPS), Optical Circuit Switch (OCS), and SDN controller. It has been integrated with *Dimemas* in order to conduct studies on performance prediction and to use realistic traffic from applications in the HPC. Additionally, it has been integrated with the InfiniBand simulator [IBSim], since InfiniBand is the most popular interconnect technology for HPC data centers (more than 40% of the Top500 supercomputers use it [7]).

2.5.1. Helper tools

In this section we introduce the simulation framework, which consists of a new simulator developed within WP2 of the project, but makes use of existing tools as well.

The general context in which the simulation framework can be employed is shown in Figure 14.

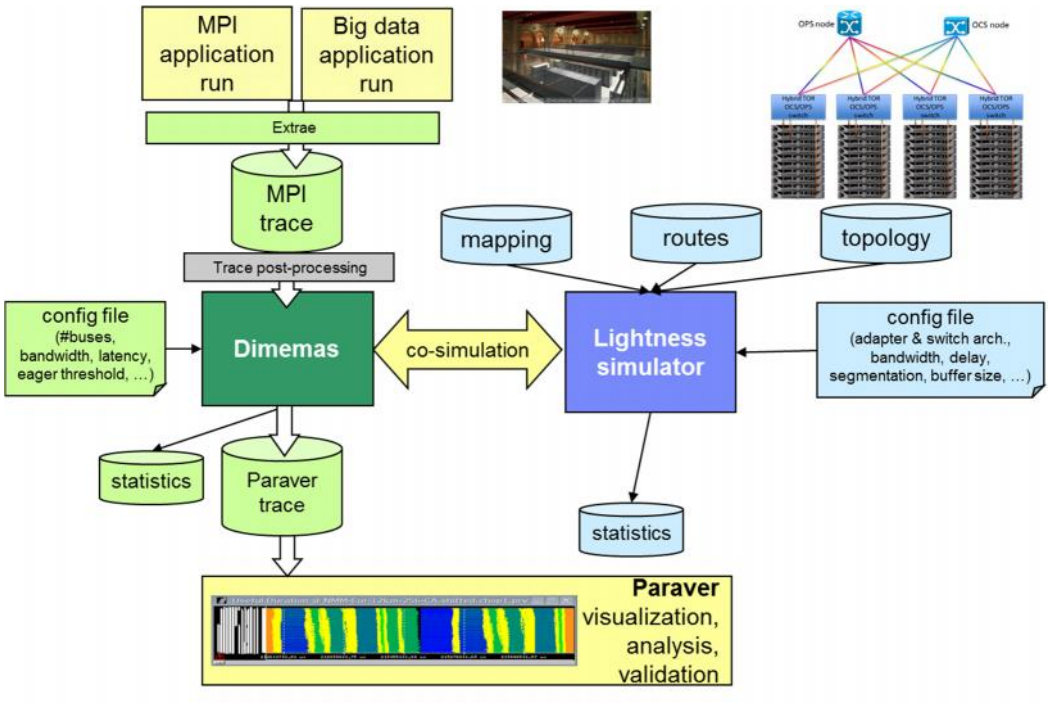


Figure 14: General context for the use of the simulation framework

Extrae [Ext] is a package developed at BSC, which can instrument applications based on MPI, OpenMP, pthreads, CUDA, etc. The information gathered by Extrae typically includes timestamps of events of runtime calls, performance counters and source code references. Additionally, Extrae provides its own API to allow the user to manually instrument the application of interest.

Dimemas [Dim] is a performance analysis tool for message-passing programs, developed at BSC. The main modelling concepts and configuration files for the tool have been described with more details in deliverable D2.2 [del-d22] of the project. Dimemas can replay traces collected by Extrae and perform prediction studies and "what-if" analysis for various system architectures, specified by the user through configuration files.

Paraver [Prv] is a visualization and analysis tool, developed at BSC as well. Paraver is very flexible and can be easily extended to support new performance data or new programming models, without changes to the visualizer. The tool offers a large set of time functions, a filter module, and a mechanism to combine two time lines, which allows displaying a huge number of metrics with the available data.

In the context shown above, Extrae is used to instrument HPC applications based on the MPI standard. The resulting traces can be further processed with Dimemas, in order to identify performance issues. However, for network communications Dimemas uses a linear performance model, and although some non-linear effects such as network conflicts are taken into account, this approach may be too simplistic. Therefore, Dimemas has been integrated with the LIGHTNESS simulator, which models network functions accurately.

2.5.2. Performance evaluation

In this section we present an experimental evaluation of the optical AoD network proposed by LIGHTNESS project and it is compared the performance with the electrical equivalent Infiniband network using different mappings and various HPC applications. In Table 2 we summarize all the parameters used in the evaluations made for this deliverable. Several HPC applications were used for this evaluation. Table 3 summarizes the details briefly of the HPC applications selected. These applications are written using the MPI parallel programming model. The applications were run in the *MareNostrum* supercomputer to obtain traces from their execution. *MareNostrum*'s nodes consist of two processors Intel SandyBridge-EP E5-2670/1600 20M 8-core at 2.6 GH with 32GB DDR3-1600 memory modules. Executions were made using 8 MPI processes in 8 different racks.

Parameters	Values
OPS Latency.	25 ns.
ACK Processing Delay.	100 ns.
Cable Delay.	5ns per meter.
NIC Delay.	300 ns.
OCS Latency.	25 ms.
SDN Delay.	512 ms.
OPS Provisioning Time.	214 ms.
Wavelength bandwidth.	8 Gbps.
Token Interchange Time.	300 ns.

Table 2: Simulation parameters

Name	Processes	Problem size	Description
MINI MD	8, 256	32 x 32 x 32	Molecular dynamics application LAMMPS [8].
MG	8, 256	256 x 256 x 256	Multi-Grid on a sequence of meshes, long and short distance communication, memory intensive [9].
CG	8, 256	14000	Conjugate Gradient, irregular memory access and communication [9].

Table 3. Selected HPC applications

Before describing the experiments made, it is important to highlight that the SDN has been also modelled in the *DimlightSim* and all devices involved in the simulation of a trace first contact the SDN in order to setup paths. In order to perform the comparisons, the first path setup time is not being taken into account. Then, when executing simulations using the OPS, the SDN impact is not observed, since the paths are just set once. However, when using the OCS switches, each time a NIC needs to make a path change, it needs to contact the SDN, break the current path and ask for a new one paying the associated delays.

Figure 15 describes the network model for one AoD cluster on the left and the equivalent model on InfiniBand (IB) networks. Note that in InfiniBand we need an InfiniBand switch to replace the one acting as a TOR and

then another IB switch to connect other racks in the system. In addition, it is also shown the fiber distance between different components in each network in meters. In both network we use the same distance. This is relevant to calculate the fiber delay in the simulator.

Figure 16 shows the performance improvement of AoD cluster with respect to IB network. Several process mapping strategies have been explored mapping all application processes in one, two, four, or eight racks. In AoD cluster all application processes are using the OPS. As it can be seen, in all cases the proposed AoD network outperforms InfiniBand network, especially for CG application. For this application up to 19% performance improvement can be seen in the case of 4-process mapping technique. The reason for this improvement is due to the fact that AoD network introduces much less delay than the IB network. Notice that in the IB network every packet has to go through the IB switch that has a considerable latency. On the other hand, in the AoD cluster the packets only incur a delay of the OPS which is substantially less than one IB switch. The obtained results encourage the utilization of AoD network in a single cluster since the applications seem to obtain benefits from the point of view of execution time. Note that the resulting number of retransmissions in these experiments has been less than 3% of the total number of packets transmitted, and thus the packet retransmissions due to packet drops were not significantly impacting performance for these applications.

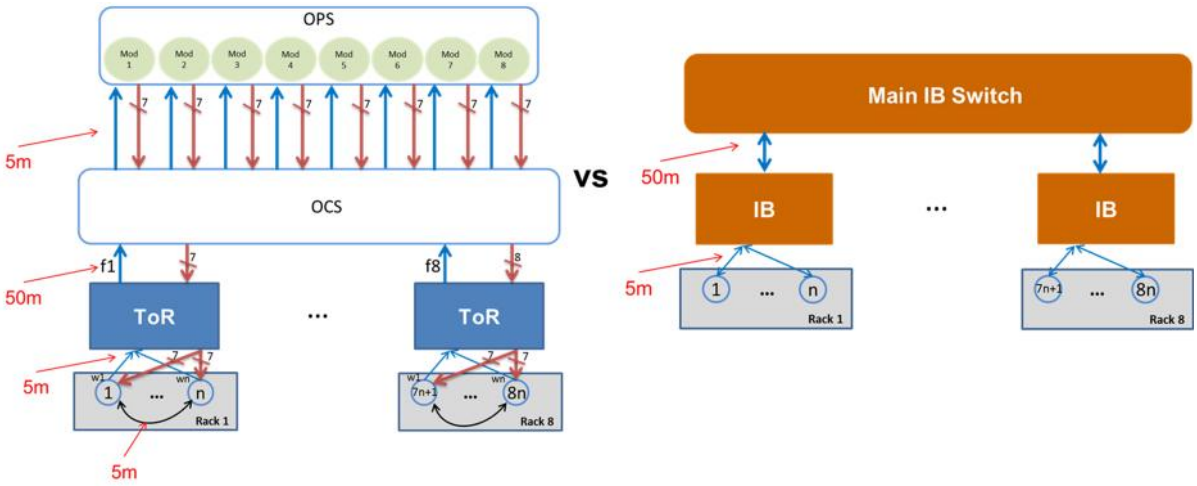


Figure 15. AoD network and its equivalent InfiniBand network

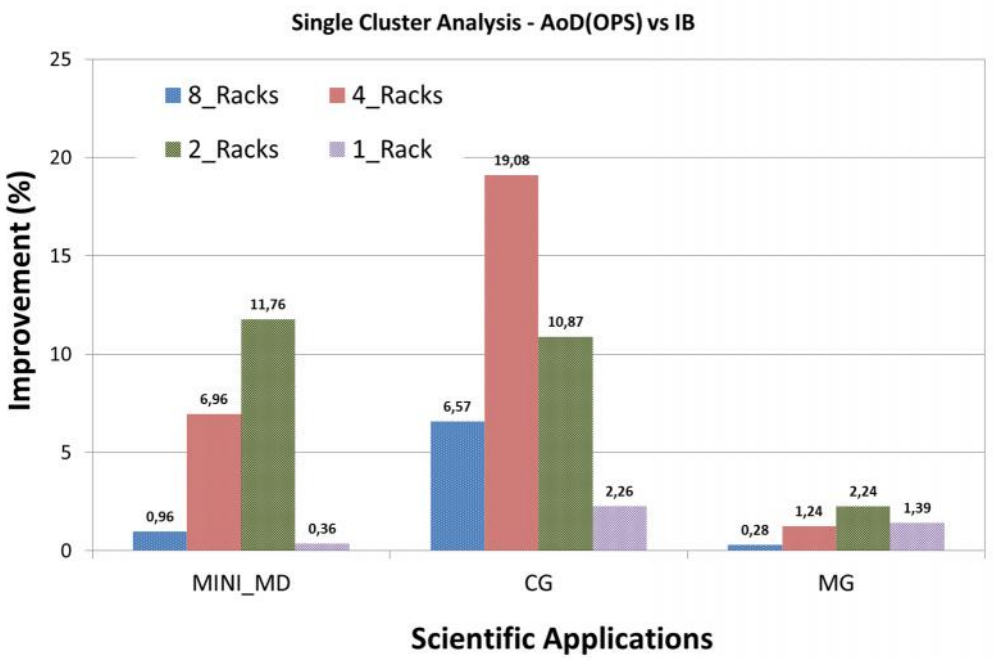


Figure 16. Process mapping analysis

Figure 17 shows the performance of HPC applications in the case of using AoD multi-clusters. In particular, the case of using two AoD clusters but varying the number of racks per cluster is considered. Note that every rack in each cluster has a single independent connection to connect to other racks in the same or other cluster. The figure on the left shows the case of using two racks per cluster whereas the figure on the right shows the result for the case of using only one rack per cluster. As it can be seen, when applications are distributed among more than two racks (figure on the left), a lot of disconnections may occur when using the OCS and this may severely impact the execution time of applications. The degradation obtained is very high, however if the applications is concentrated in two racks, we still are able to obtain improvements for all the analysed applications because once the OCS setup the path between these two racks located in different clusters at the beginning there is no need to do more connections in the OCS and thus the performance is much better than the case of IB networks because the delay to go through the OCS is much less than the delay of IB switches.

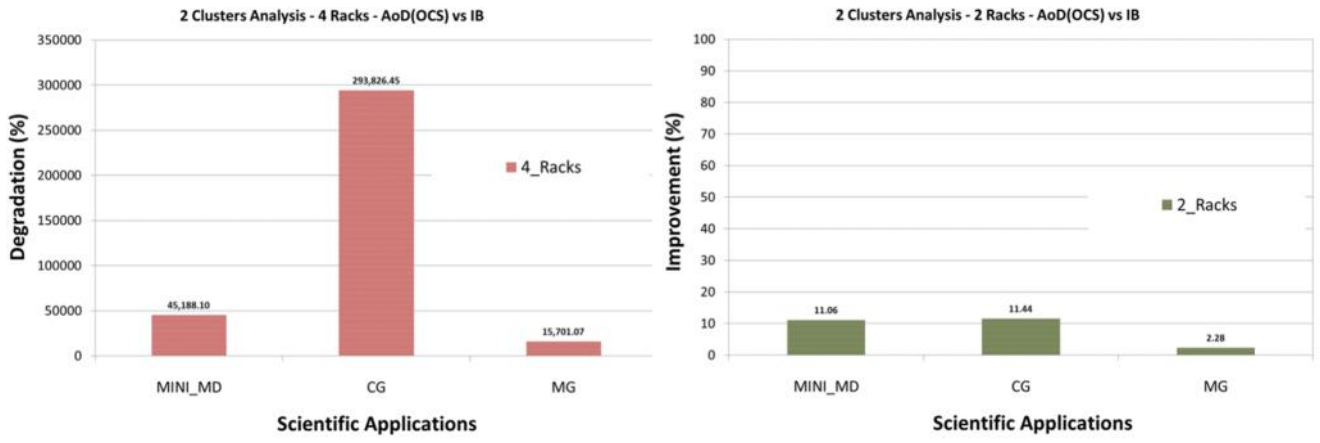


Figure 17. Multi-cluster performance analysis

3. Scaling AoD network

This section investigates different approaches to scale up the AoD network architecture to a large number of servers. The first approach is focused to scale up the number of racks in one AoD cluster. The second approach is focused on scaling up the number of servers in a rack and thus the total number of servers in one AoD cluster. These two approaches are promising techniques to increase the number of servers in an AoD cluster. Finally, it is provided a scalability study on the characterization of HPC workloads on larger problem sizes. This will provide insights on the potential performance impact of HPC on optical devices on large executions.

3.1. Scaling racks

The size of the OCS determines the maximum size of the OPS that AoD network could be supported. As it was calculated in Section 2.2, the size of the OPS is following a quadratic relationship with the size of the OCS. Specifically, $N_o = 2 * O^2_{in/out}$ determines the number of ports needed in the OCS in order to connect an OPS of size *input*. Because of that, one large OPS ends up to be requiring lots of ports in the OCS. For example, a 9-port OPS would need 162 ports in the OCS. Notice that a 9-port OPS could only connect 9 racks in the system. Imaging that it is not used at all an OPS then 162 ports could be used to connect 81 racks instead of only 9. However, the downside is that racks could only connect to the OCS and not OPS. As you can see, a large OPS is consuming lots of ports in the OCS that could be used to support a higher number of racks in the system due to the quadratic relationship. In this section, it is proposed to use smaller number of OPS in order to support a larger number of racks without penalizing the advantage to use OPS. It is still guaranteed that racks could use OPS, but without requiring a large number of ports in the OCS. Smaller OPS could still connect the same number of racks than a larger OPS could support. The only drawback of this approach is that it is limiting the number of racks that could simultaneously use the same OPS. Therefore, it could limit the size of applications that use OPS. Notwithstanding, this approach will support a much larger number of applications running concurrently in the system which is essential for capacity data centers.

In order to illustrate this technique, Figure 18 shows the number of racks that could be connected to the OPS in the case of using a 6-port OPS instead of the 9-port OPS when using a 192-port OCS. Green rectangles show the racks that are connected to the OPS and blue ones show racks connected to the OCS. As can be seen, it is now supported two 6-port OPS instead of a large 9-port OPS. This small change makes big improvements on the AoD network because now the number of racks has been increased to 60 from 39 that were supported in the 9-OPS case.

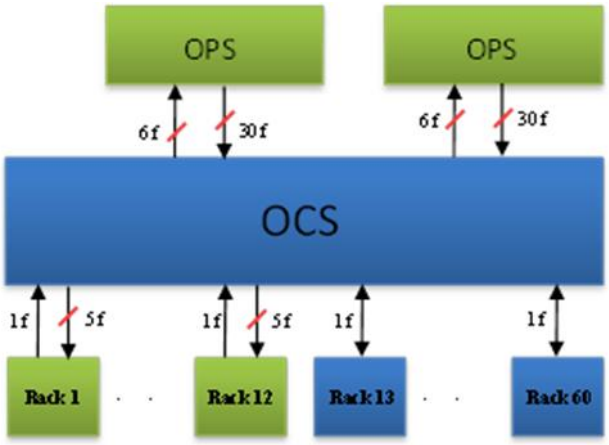


Figure 18. Using 6-input port OPS switches

In order to quantify the number of racks that this approach could support a model has been developed following the equations presented in Section 2.2.

Taking into account Equation 2 in Section 2.2, it is necessary to have 2 x N_{OPS}^2 available OCS ports to connect OPS to OCS. Knowing this, the number of OPS switches (M_{OPS}) is possible to connect to an OCS switch of limited number of ports in the OCS (N_{OCS}), which could be easily calculated by:

$$M_{O\ S} = \frac{N_O}{2\ x\ N_O\ ^2} \quad (15)$$

Where N_{OPS} is the number of input ports in the OPS. The number of racks that could use OPS switches (R_{OPS}) is given by:

$$R_{O\quad} = M_{O\quad}\ x\ N_O \quad (16)$$

It takes into account only the OPS switches that are the same size. It takes just the integer part of M_{OPS} value. If M_{OPS} has a decimal part, it represents the remainder which can be fulfilled with the smaller size OPS switch or with connections of racks that will communicate only through the OCS.

Figure 19 shows the resulting number of racks that connects to OCS and OPS when using various OPS sizes. It is assumed a 192-port OCS is been used. The maximum number of racks which can use OPS switches (red rectangles) while the remaining ports are assigned to racks that will use only OCS (green rectangles). As can be seen, more than a hundred racks are supported in the case of 7-port OPS.

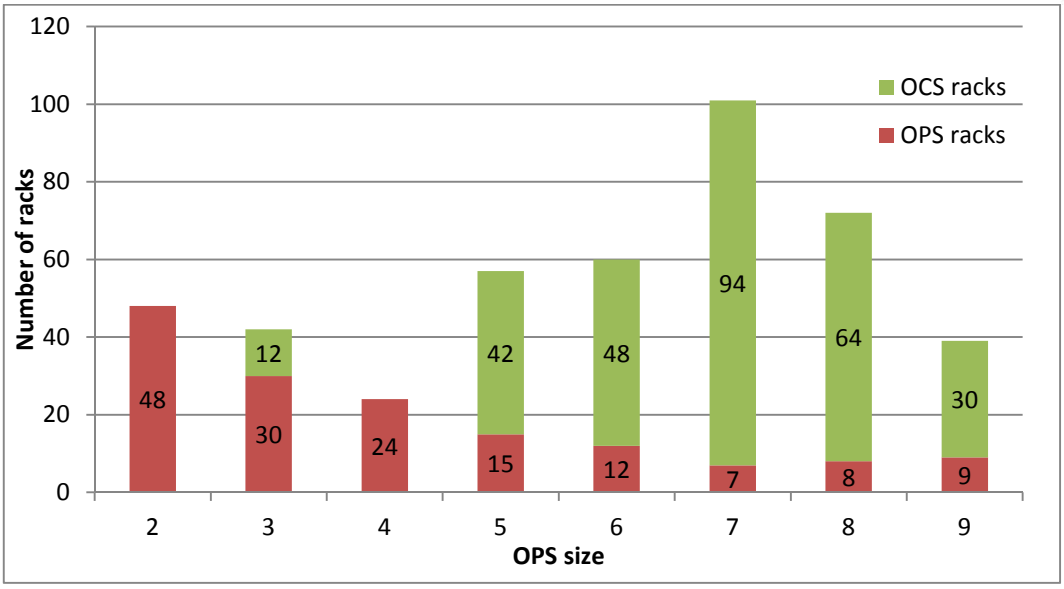


Figure 19. Number of rack supported when varying the size of OPS.

3.2. Scaling servers

Taking into consideration how the data centres and HPC are growing, it is important to explore techniques that allow connecting more servers or NICs to switches without necessarily increasing the port-count of switches. Using one fiber per each NIC could be the best solution if we are using OPS and if we take into account just performance. This is because the number of collisions would be zero, since each server will have its own fiber. However, in terms of scalability, this solution cannot be applied because only 1 wavelength of each fiber is being used. Figure 20 shows this scenario, where the main problem is that as the OPS receives input from nine fibers (*F*), it will has as output 72 fibers (*F* x (*F* - 1)) that connects it with the OCS.

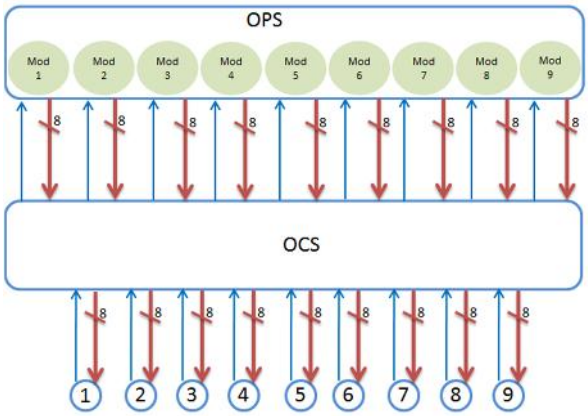


Figure 20. 9-server AoD network.

When using DWDM several wavelengths can be multiplexed over a fiber, and if one wavelength is assigned to each NIC, we can reduce the number of ports used by OCS and OPS devices. Figure 21 shows how ToRs may be used to aggregate the traffic from *n* servers, each one using one wavelength. Having a total of 9 fibers up to 9*n* servers can be plugged to a single OCS when the OPS being used on top of it.

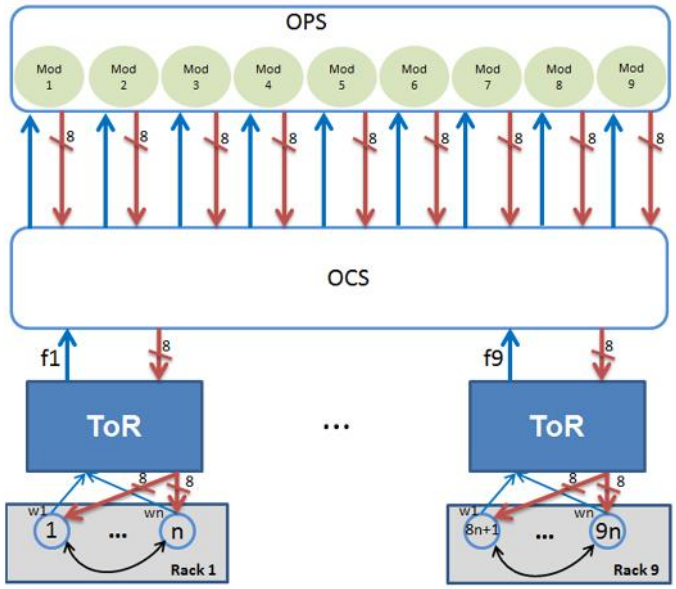


Figure 21: Using the ToR to aggregate traffic from different wavelengths into one fiber

By using DWDM the number of servers per port can be increased, however there is a physical limitation in the number of wavelengths that compose a fiber (current commercial fibers support around 80 wavelengths).

Taking into account the aforementioned limitation we have designed a methodology based on Time Division Multiplexing (TDM) which relies on the use of a Combiner that is composed by a Multiplexer (MUX) and a Splitter. Notice that this solution assumes that there is no limitation on the number of ports in the NIC in order to connect using full mesh all the NIC in the rack as defined in the AoD network.

Figure 22 shows how the Combiners are used to increase the number or servers per ToR. Up to *p* servers are connected to each Combiner, and each server behind each Combiner uses the same wavelength. Each combiner outputs a single wavelength and all the input wavelengths are mixed in one single wavelength. The proposed hardware for the Combiners is a combination of Multiplexer (MUX) and Splitter.

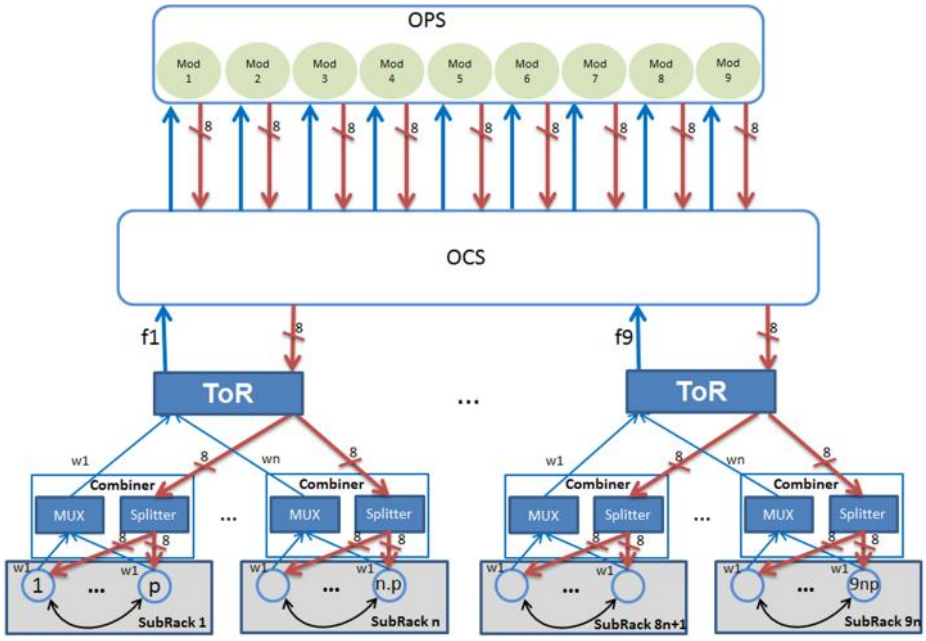


Figure 22. Using the Combiner to increase the number of servers using TDM.

Comparing Figure 21 and Figure 22, it can be seen that in the second approach the number of servers can be increased from 9*n* to 9*np* where *p* is the total number of servers connected to each Combiner. The Combiners give the possibility of going beyond the physical limitation of optical fibers.

In order to allow the combination of *p* servers that use the same wavelength, we have designed an adaptive token mechanism that is used inside the NICs in order to serialize the access to the Combiners. Algorithm 1 describes the token mechanism implemented inside the NICs. The token circulates between servers that are in the same rack and servers can only transmit packets when they have the token. It is important to highlight that packets discarded by the OPS are queued again in the *SendingQueue*, so they can be retransmitted before other packets.

Figure 23 shows an example of the adaptive token mechanism that we have designed. It can be seen that the period of time that a NIC holds the token could vary depending on the case. Assuming that the Token Hold Time is 300 ns, NIC 0 holds the token during 300 ns, because the packet that it has to transmit takes only 200

ns to finish. On the contrary, NIC 1 has a large packet that takes 400 ns to finish its transmission, then it holds the token until the message transmission ends. NIC 2 does not have anything to transmit, so it gives to token right away and finally NIC 3 holds the token for 300 ns, and it transmits a packet that takes 100 ns.

```
Data:
Token Time = T_Time;

begin
    WaitForToken();
    if SendingQueue:isEmpty then
        TransmitToken(NextNeighbor);
    end
    else
        CurrentT okenTimer = StartTokenTimer(T_Time);

        while CurrentTokenT imer > 0 do
            PopPacket = getPacket(SendingQueue);
            TransmissionTime = SendMessage(PopMessage);
            WaitTransmission(TransmissionTime);
            CurrentTokenT imer = CurrentTokenT imer – TransmissionTime;
        end
        TransmitToken(NextNeighbor);
    end
end
```

Algorithm 1. Pseudo-algorithm of the NIC's token mechanism

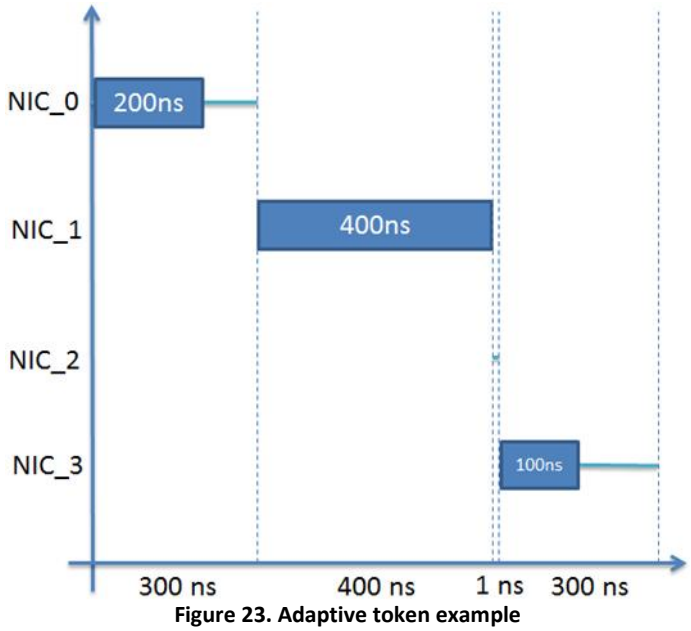


Figure 23. Adaptive token example

3.2.1. Simulation design of the Combiner

In Figure 24 are described the main components of the Combiner and how it is connected with the NICs and the OCS switch at the simulator level. The Input module manages inputs from different NICs and forwards the packets to Control OCS module, which is in charge of sending and receiving packets to/from the OCS switch and also receives the ACKs generated by the OPS switch. The Control OCS also forwards received packets to the Output module, which is in charge of selecting the proper output port to reach destination NIC.

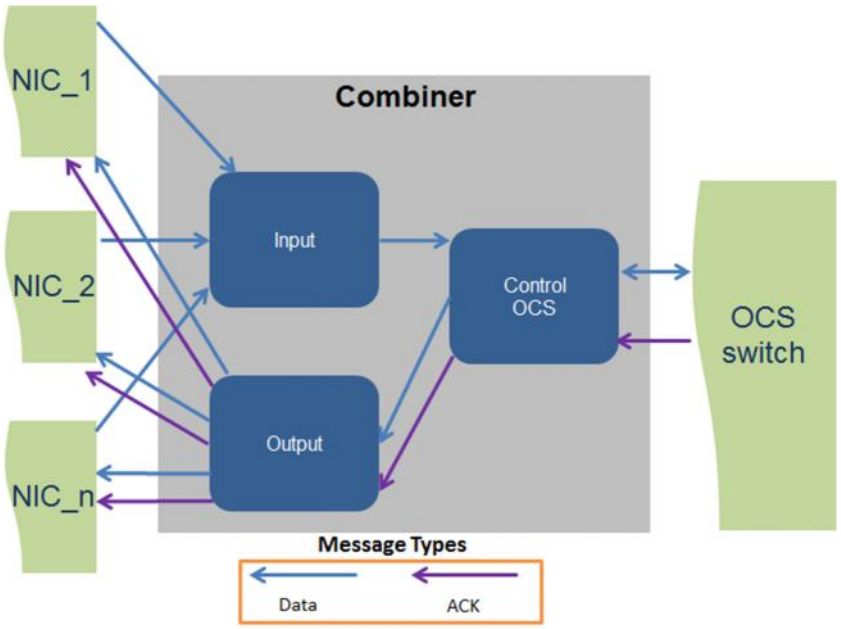


Figure 24. Logic modules of the Combiner

Figure 25 depicts the main components and interactions in the NIC. The Input module receives inputs from the Server and if the packet is going to a server in the same rack, it forwards the packet to the corresponding Control Intra module. If a packet is going outside the rack, the Input module forwards the packet to the Control Inter module which converts the packet to photonic and transmits it through the corresponding port. When using the OPS and a collision occurs, the Control Inter receives a NACK and resends the packet. When the Control Inter receives a packet to the server connected to it, it transmits the packet to the Output module. It is also important to highlight that the Control Inter has a port that is used to connect it with the SDN device. When the Control Inter receives a packet, it checks if there is a connection available to reach the destination NIC, if not it asks the SDN to set a new path to the destination.

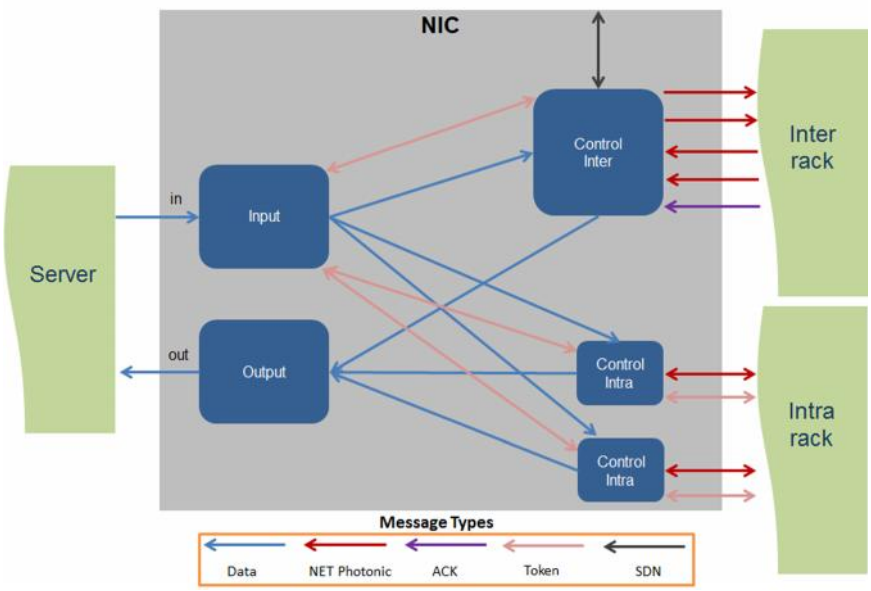


Figure 25. NIC Logical Modules

When the Combiner is being used, the token mechanism is activated inside the NIC. Each rack has a NIC that is in charge of initiating the token mechanism called Token Initiator. The Token Initiator maintains the token during a period of time (T_{Time}), and during this period it can transmit packets to servers in another rack through the Combiner. Once the T_{Time} expires, the Control Inter sends the token to the Input module which selects the next NIC in the ring that has to receive the token and transmit it to the corresponding Control Intra module.

3.2.2. Experimental evaluation of the Combiner

In this section, we also present experiments that show the performance of scientific applications on an interconnected network based on the AoD explained previously. For these experiments we consider that all servers are connected to an OCS and they transmit their packets making use of the OPS that is also connected to the OCS. The main objective is to analyse the impact in performance when using the techniques presented to reduce the number of resources used: ToR and Combiner.

In these experiments, we are not considering the usage of ToRs and Combiners at the same time, as shown in Figure 22. We evaluate separately the DWDM technique using ToRs and the TDM technique using the Combiners.

The selected applications for these experiments are described in Table 3. All the applications were run in the MareNostrum supercomputer to obtain traces from their execution. MareNostrum's nodes consist of two processors Intel SandyBridge-EP E5-2670/1600 20M 8-core at 2.6 GH with 32GB DDR3-1600 memory modules. All applications run on 8 and 256 processes. Executions with 8 processes were run using 8 different racks and executions with 256 processes were distributed among 16 racks.

Once the traces were obtained, the *DimLightSim* was used to simulate the effects of different network configurations and mappings on the parallel applications.

Table 2 shows the parameters used for the simulations executed with *DimLightSim*. These parameters were taken from measurements from the real optical devices. The Token Interchange Time was set to 300ns, since experimental tests made showed that this value allows the transmission of packets of an average packet size. Nevertheless, we will address in the future the challenge of selecting an ideal Token Interchange Time. For simplicity, we are assuming that the buffer size in the NICs used for packet retransmissions is infinite. The evaluation of the impact of limited buffer size will be addressed in a future work. In the latter case, more packets could be dropped as the NIC buffer becomes full.

In all the experiments that are presented here, one fiber is used to connect each rack to the OCS. It is important to highlight also that when using the OPS, the number of output fibers increases almost quadratically, as was explained before.

The main goal of this experimental evaluation is to demonstrate that the Combiner and the Adaptive Token Mechanism do not seriously impact the performance of parallel applications while allowing to increase the number of servers that can be connected to an OPS.

Figure 26 shows the simulation setup used to compare the ToR and the Combiner using as the base case Infiniband. As it can be seen, we have used a Main IB Switch that acts as the OPS and Cluster IB switches in order to allow switching packets inside the same cluster, in order to mimic the behaviour of intra-cluster communications made through the NICs. The different lengths of cables are specified in

Figure 26 and different mappings were used to analyse the impact in performance of HPC applications.

Figure 27 shows the obtained results when distributing applications processes among 4 and 2 different racks. As it can be seen, in all cases (using ToR and Combiner) the AoD proposed is able to outperform IB. Depending on the applications, the improvement in execution time varies between 1.21% and 19.08%. It is important to highlight that the performance improvement obtained when using the combiner is quite similar to the obtained when using the AoD ToR, even considering the use of the TDM mechanism described previously. The difference between the AoD ToR and the combiner varies between 0.03% (MG-4 racks) and 0.68 (CG-2 racks).

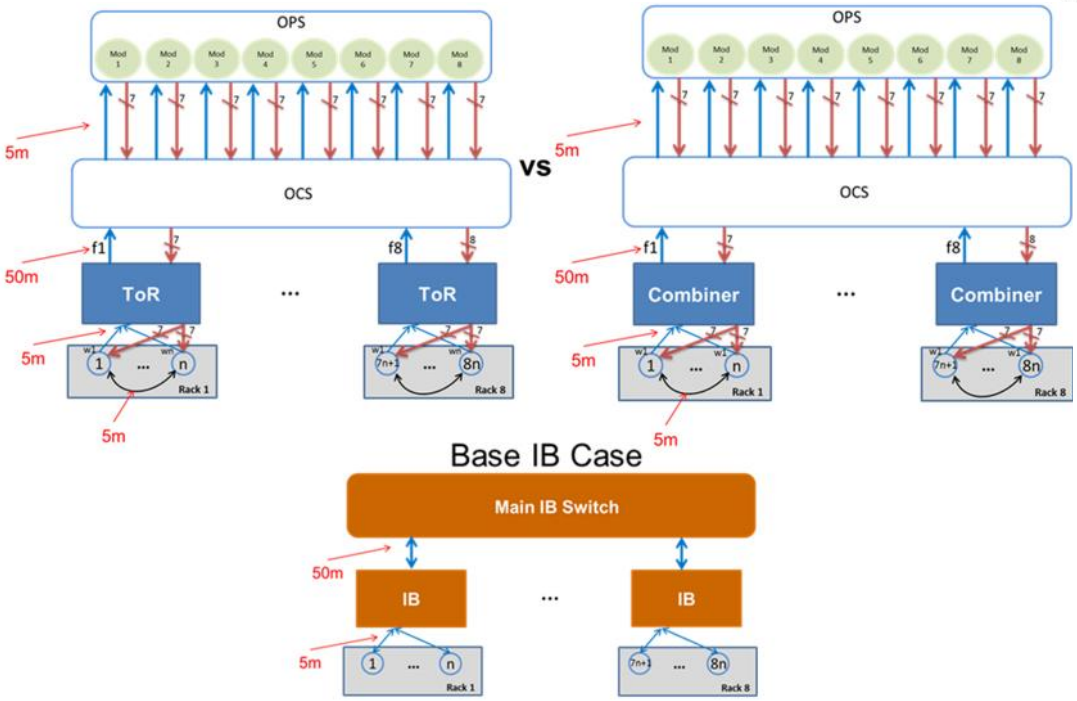


Figure 26. Simulation setup to ToR and Combiner performance with respect to the base IB case

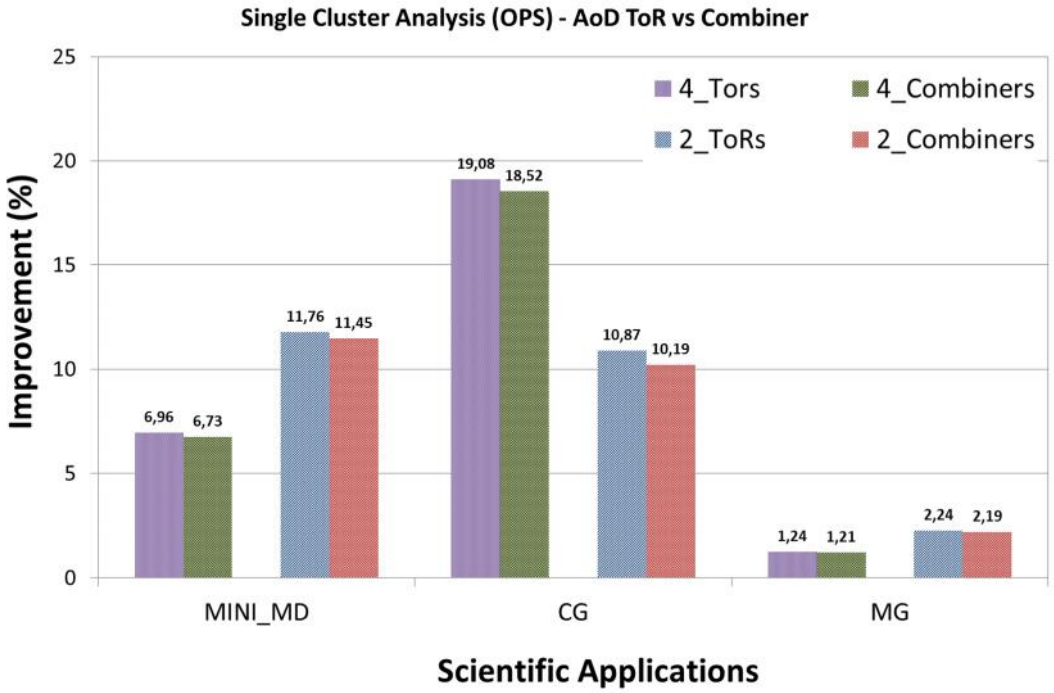


Figure 27. Performance comparison between AoD ToR and the proposed Combiner using IB switching as base case and 8 processes

Figure 28 depicts the percentage of improvement obtained when ToRs and Combiners are compared against Infiniband switching using 256 processes distributed among 256 servers. As it can be observed in all cases, the AoD using ToRs and combiner outperforms IB (between 13.49% and 69.61%). Processes were distributed

among servers located in 32 and 2 different racks. ToRs and Combiners where used to connect the racks with one OCS. Taking into account the obtained results, the usage of Combiners can lower the improvements in execution time between 2.45% (MINI_MD – 32 racks) and 20.35% (CG – 32 racks). However, for the specific case of the MG application, it can be seen that the Combiner outperforms the ToR since the retransmissions made when using the ToRs have a higher impact in the execution time than the TDM mechanism. The number of packet retransmissions are 0.27% for the MG with 32 ToRs and 3.6% for the MG with 2 ToRs.

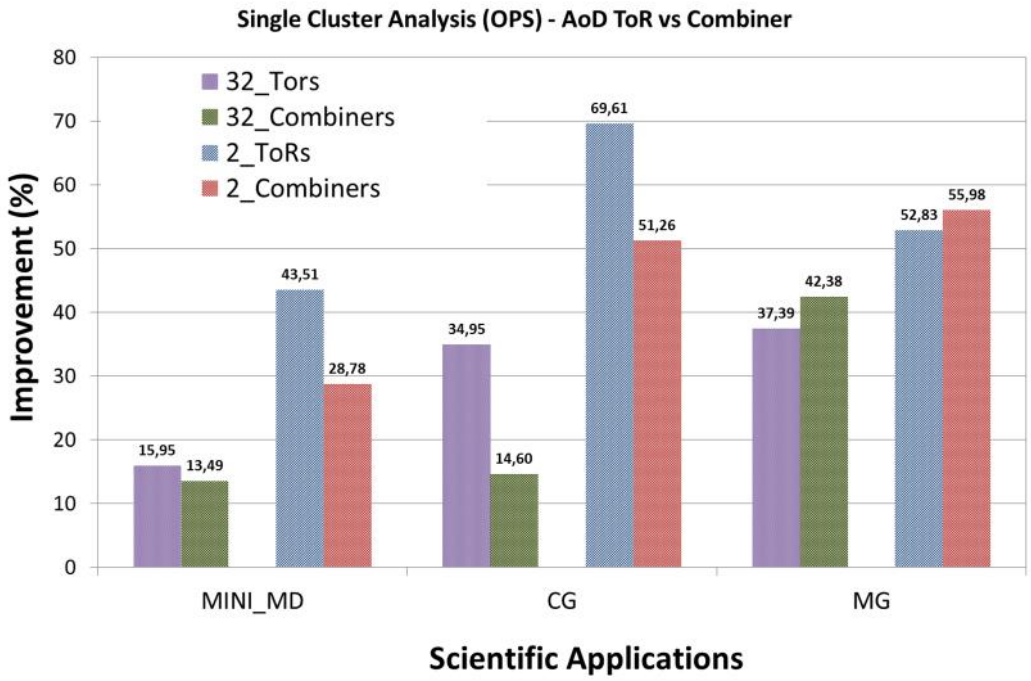


Figure 28. Performance comparison between AoD ToR and the proposed Combiner using IB switching as base case using 256 processes

The obtained results presented in this section encourage the usage of the Combiner in order to increase the number of servers that can be accommodated per fiber, since several servers could share the same wavelength. However, it is important to analyse the trade-off relationship between performance improvement and scalability, since when increasing the number of servers per wavelength the impact in performance should be taken into account. As a future work we will focus on analysing the appropriate token interchange time according to application characteristics and we will analyse the appropriate number of servers that can share the same wavelength (*p*).

3.3. Workload scalability

The impact of increasing the problem size in the application performance when using an OPS or OCS in the network will be analysed in this section. A larger problem size could have a significant impact in the concurrency and the sensitivity of applications to OPS collisions resulting in a higher performance impact. However, this will strongly depended on the application behaviour as it will be shown below.
Table 4 shows the different workloads selected for each HPC application analysed. Problem sizes range from small (default) to large. It also shows the duration of the trace for each application. The same was taken for each application on every problem size in order to properly analyse its impact.

Application	Small	Medium	Large	Time (ms)
HYDRO	250x125	500x250	1000x500	4.8
MILC	16x16x16x16	16x16x16x32	16x16x32x32	0.36
MINIMD	32x32x32	64x64x64	128x128x128	254.8
SNAP	4x4x4	16x16x16	32x32x32	1.8
MG	256x256x256	512x512x512	1024x1024x1024	428.3
CG	14000	75000	1500000	4.9

Table 4. HPC applications size and execution time

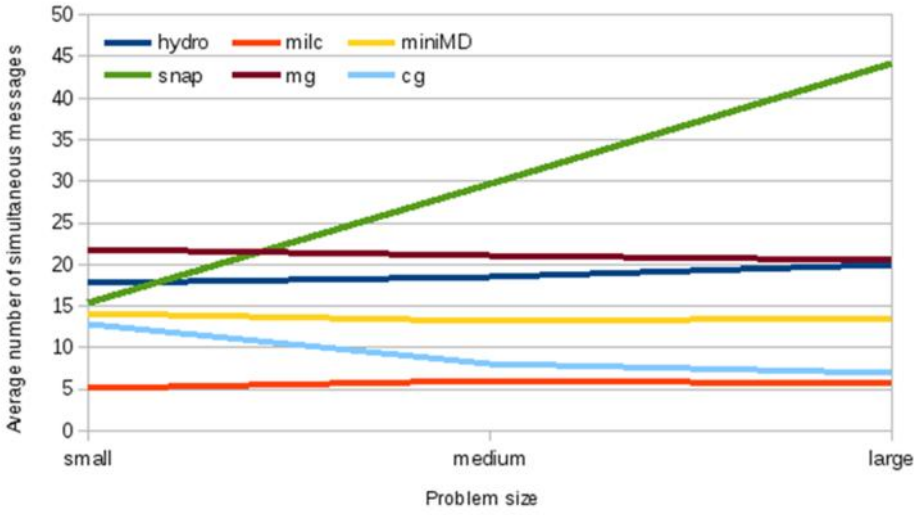


Figure 29: Concurrency for different problem sizes

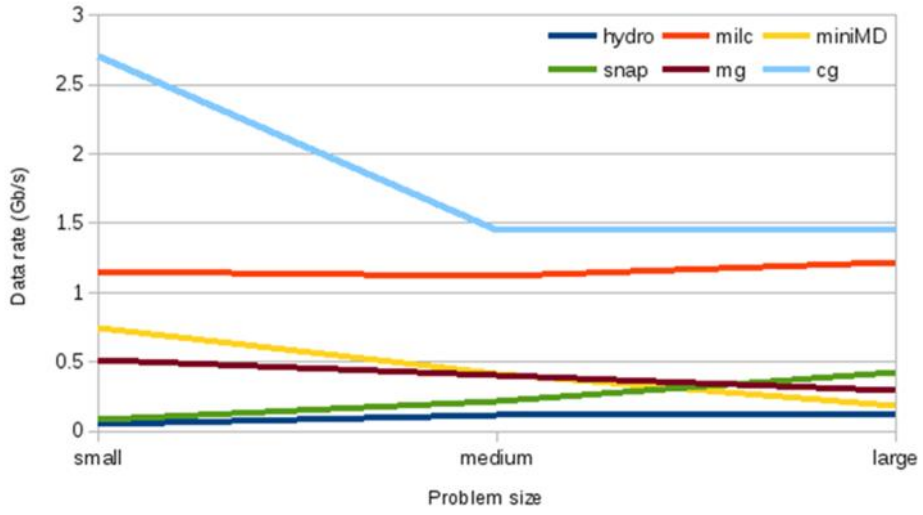


Figure 30: Data rate for different problem sizes

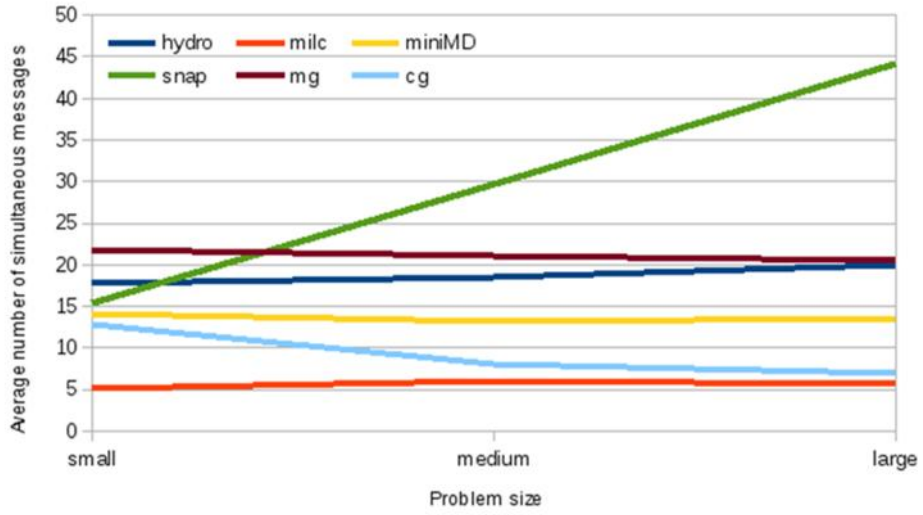
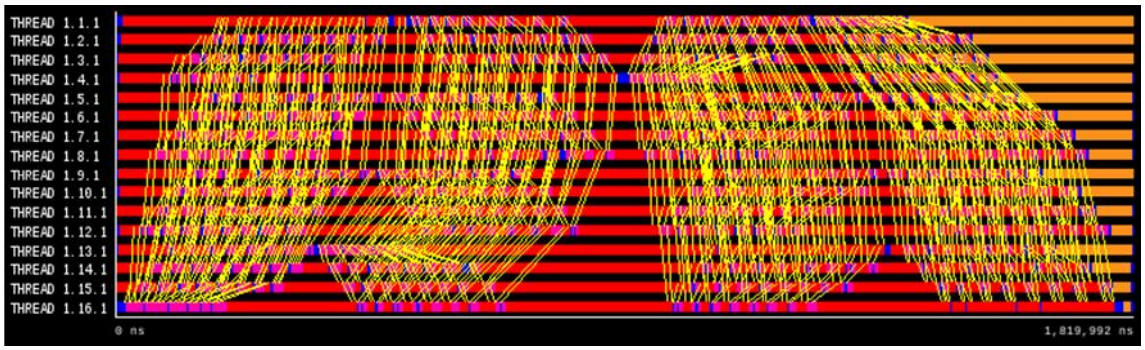


Figure 29 shows the concurrency and Figure 30 shows the data rate obtained for different problem sizes for each HPC application. As it can be seen, the variation in the problem size impacts differently on each application' performance. Some applications exhibit a significant concurrency increase and also data rate such as SNAP, MILC, and HYDRO whereas others exhibit a decrease of these metrics like CG, MINI_MD, and MG. In addition, a correlation can be seen between concurrency and data rate, when concurrency is decreasing the data rate is also decreasing, and vice versa.

The significant increase of concurrency and data rate of SNAP is due to the fact that this application sends more messages as the problem size is increased, and thus both the concurrency and data rate are increased. This effect is illustrated in Figure 31.

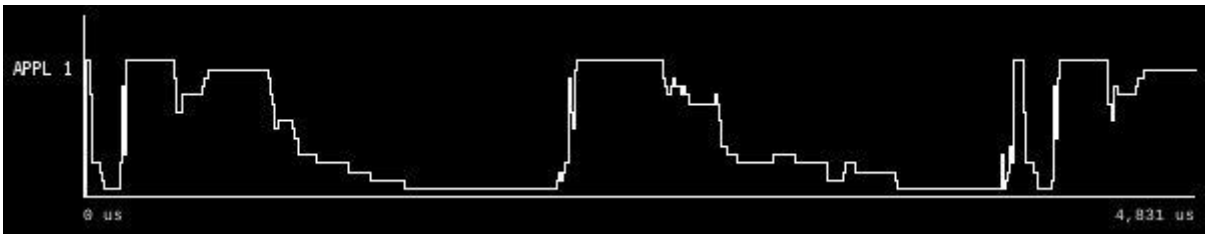
(a) *SNAP* small problem size



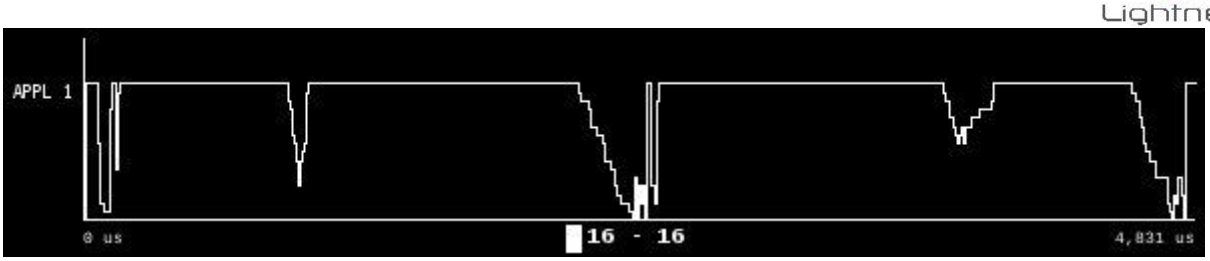
(b) *SNAP* large problem *size*

Figure 31: Trace of the application SNAP for different problem sizes

Another cause for the increase of the concurrency is due to the fact that the application gets more synchronized when executing larger workloads, as it happens with the HYDRO application. Figure 32 shows instantaneous parallelism of the HYDRO application over time, so the effect of synchronization can be observed. This metric shows when the application's processes are communicating. When they communicate at the same time, then the instantaneous parallelism is decreased. As it can be seen, with larger domains the computation increases significantly and the communication is placed in shorter time interval. The communications between servers are more synchronized, increasing the level of concurrency.



(a) HYDRO small problem size



(b) HYDRO large problem size

Figure 32: HYDRO instantaneous parallelism for different problem sizes

On the other hand, the data rate could decrease for some applications such as the MINI_MD because the computation time increases significantly with bigger workloads. Figure 33 shows the time that this application spends on computation and communication for different problem sizes. As expected, the application's computation is increased significantly when the problem size is increased. The communication time also increases, but it is not considerable when taking into account the computation. This behaviour is also expected because these applications only communicate the surface of the problem volume, and thus the surface is increased less than the volume when the problem size increases. It is important to highlight that the message size in large problem sizes is also increased as shown in the following Figure 34.

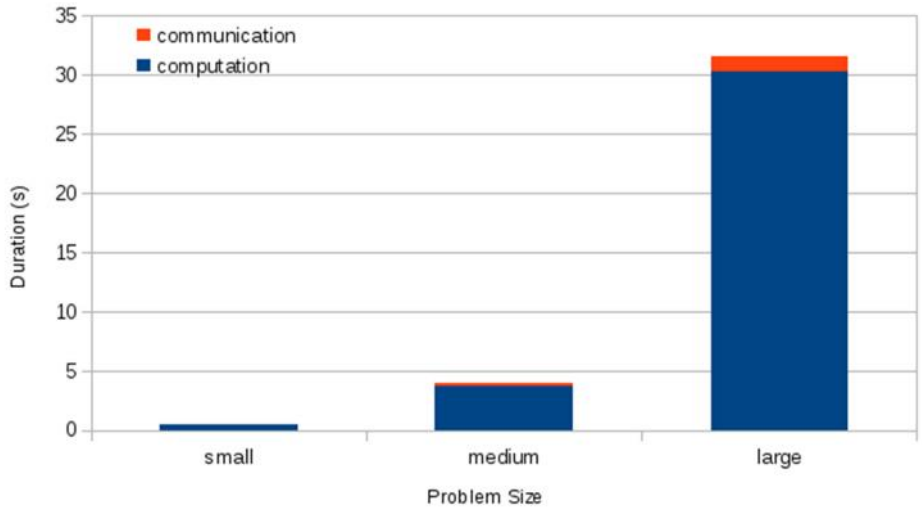


Figure 33: Time between communication and computation for MINI_MD for different problem sizes

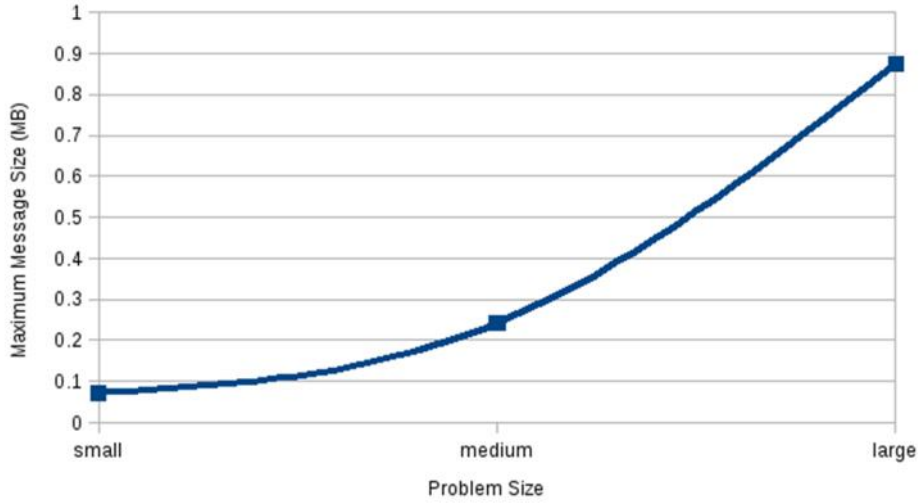


Figure 34: Message sizes of the MINI_MD with different workloads

When analysing the impact in performance when using the OCS switch, it is very important to analyse how many paths the application processes need to create during the execution. Many paths would lead to higher delays, because time cost for every path setting is 25ms. Figure 35 shows the percentage of OCS changes over the total number of sent messages. Bigger workloads show similar amount of destination changes. This characteristic shows that the nature of the analysed applications doesn't change when increasing the workload.

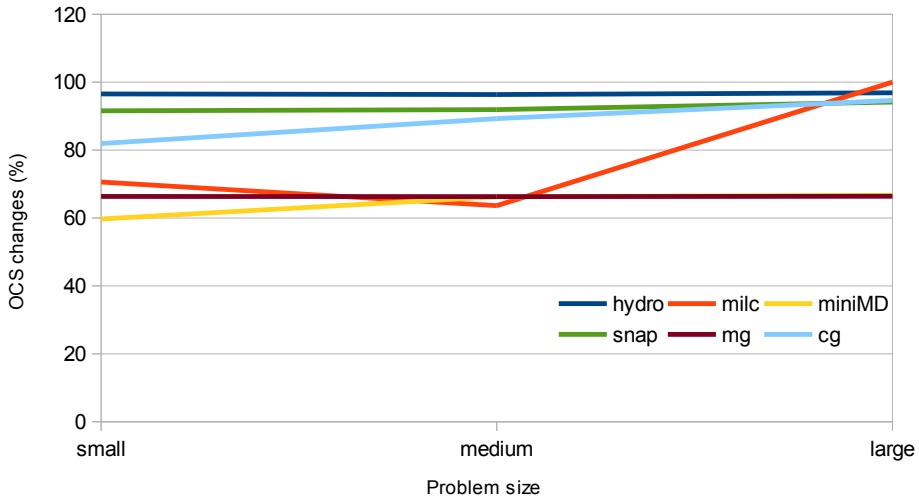


Figure 35: OCS Changes – different problem sizes

Some applications show a higher percentage of destination changes. It should be considered that the duration of the applications is the same, but they contain different number of messages. Regarding larger domains of

these applications it wasn't possible to collect all different communications in a limited duration. The larger domains might not contain all the different parts of the communication. For the *MILC* for example, the analysed part of the largest problem size has only the communication where the path for each message needs to be set. However, the conclusion is that communication of larger problem sizes is just repetitive version of smaller problem sizes, so the percentage of OCS changes stays almost the same.

4. Economical cost analysis

In this study, we investigate and compare the cost of the LIGHTNESS DCN architecture adopting optical switching technologies with current DCN architectures based on electrical switching technologies. The comparison results indicates LIGHTNESS serves as a promising solution for the DCNs offering significant less cost compared with electronic architectures based on tree-like structures (Tree, FatTree, Leaf-Spine, and super Leaf-Spine) that are typically employed in current DCN.

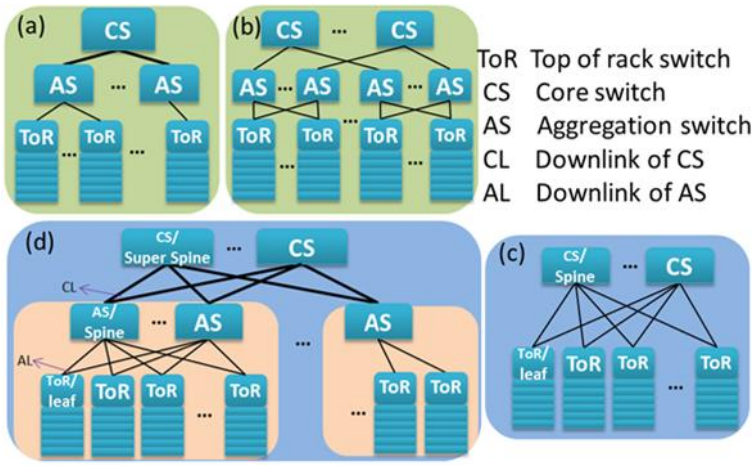


Figure 36: Electronic Data Center network architectures. (a) Tree (b) FatTree (c) Leaf-Spine (d) Super Leaf-Spine.

Figure 36 shows the Tree, FatTree, Leaf-Spine, and super Leaf-Spine network architectures that are typically employed in current DCN. The Tree DCN architecture shown in Figure 36 (a) includes access, aggregation, and core layers. The high radix switches elements are equipped in the higher layers. The FatTree architecture shown in Figure 36 (b) is composed of k pods, and each pod has k identical switches organized in two successive layers. The switch in access layer interconnects k/2 servers while the aggregation switch is connected to k/2 access switches and k/2 core switches, respectively. Thus the total servers can be supported is k3/4. Leaf-Spine [10] shown in Figure 36 (c) is widely employed in DCNs because of its flexibility to adapt to the required bandwidth. The leaf switches served as access layer are meshed to all the spine switches, which are essentially the core switches with high-throughput and high port density. The flatness of the Leaf-Spine architecture leads to a lower latency with respect to the Tree and FatTree for small size DCs. A variation of the Leaf-Spine is the super Leaf-Spine architecture [11] shown in Figure 36 (d). This includes an additional super spine layer to interconnect larger amount of leaves at the expense of extra latency and cost.

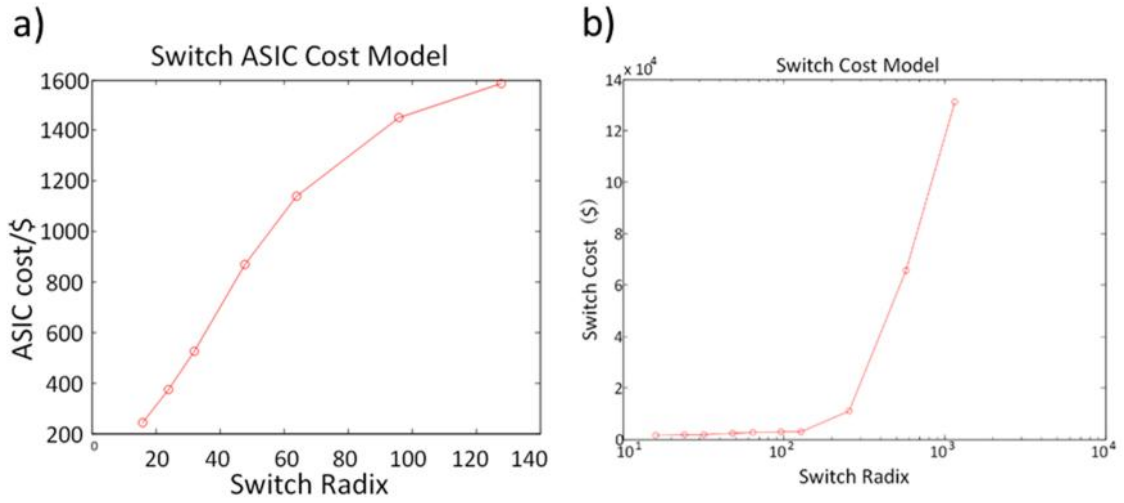


Figure 37. Switch cost analysis

To calculate the cost of the electrical switch as function of the switch radix, we break down the cost of the application specific integrated circuit (ASIC) and the mainboard. The costs of the ASICs for switches with radix up to 128 are shown in Figure 37a (Broadcom BCM56850 Series [12]). The cost of the rest components (fans, PHYs, etc.) contribute to around 400$ while the cost of the mainboard is about 1000$ from the discussions with industry experts [13]. Switch with radix up to 128 consists of a single ASIC, while the 256-radix switch is built upon six 128-radix ASICs on one mainboard adopting CLOS topology. Due to the limited space and power dissipation in one mainboard, switches with radix larger than 256 are built by multiple mainboards. For example, six 256-radix switch mainboards are used to build one 512-radix switch and so on for larger radix switch. The switch cost is shown in Figure 37b. The cost increases slowly for radix smaller than 128, while it increases rapidly when multiple mainboards are used to build high radix switch.

For the architectures discussed in this work, all the switch ports are equipped with optical transceivers except for the downlink ports from the ToR to the servers. The large amount of optical transceivers along with the optical cables used for the interconnections accounts for the major contribution to the cost of components, as shown in the Table 5. The cost of the electrical cables used for the downlinks of the ToR to the server is also included in Table 5. The listed values are based on the quotes from the vendors [14] [15].

Components		Cost($)
SFP		120
QSFP		339
CXP		650
Electrical cable	5m	13
	30m	50
Optical fiber		82
ASIC Radix	26	410
	48	870
	64	1140
	128	1587

Table 5: Component cost

Configurations \ Architectures		Tree	FatTree	Leaf-Spine	Super Leaf-Spine
Access switch (ToR)	Radix	128	128	256	128
	Amount	1563	1640	1024	1600
Aggregation switch	Radix	512/10	128	-	288
	Amount	224/16	1640	-	400
Core switch	Radix	1024	128	1024	512
	Amount	1	820	128	25
Aggregation layer oversubscription		7/14	1	-	7
Aggregation switch downlink bitrate		40Gb/s	10Gb/	-	10Gb/s
Core switch downlink bitrate		100Gb/	10Gb/	10Gb/s	40Gb/s

Configurations \ Architectures		LIGHTNESS
ToR	Radix	128
	Amount	1600
OPS	Radix	32×32
	Amount	100
OCS	Radix	192×192
	Amount	50
Cable latency		5 ns/m
OCS bitrate		100Gb/s
OPS bitrate		160Gb/s

Table 6: Architectures configuration for scaling the DCN to 100,000 servers

To compare the two technologies, electrical and optical, we consider scalable Data Center Networks up to 100,000 servers. The bitrate of the links connecting the servers, the ToRs, aggregation switches, and core switches, as well as the amount of switches, cables, transceivers, etc. for each architecture is reported in Table 6.

The cost of the InP based OPS has been calculated according to JePPIX roadmap [16], which predict a cost of the InP in the order of 100$/mm2 down to 10$/mm2 for volume from 10,000 to 100,000. It is estimated that 800mm2 is required for implementing the OPS (32×32). This leads to a conservative estimated cost of 80,000$ (800mm2×100$/mm2). According to the price of the commercially available products of Polatis, the cost of OCS (192×192) is around 340$ per port [3].

The overall cost of the DCN architectures with the individual contribution of the network is shown in Figure 38. The FatTree is the most expensive one due to the large number of transceivers that count for more than 50% of the total cost.

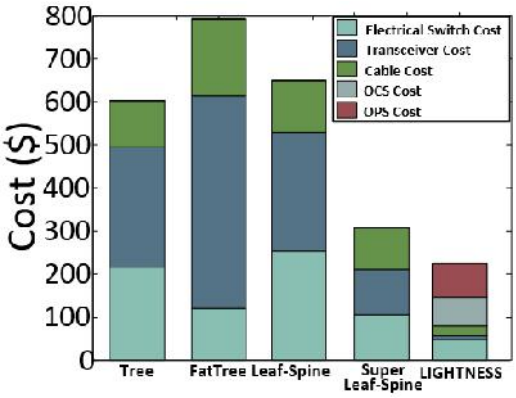


Figure 38: Cost normalized by number of servers

While the LIGHTNESS has the smallest cost resulted from the adoption of optical switches which significantly reduce the number of transceivers. The major expenditure for LIGHTNESS will be the cost of switches (~85%). As can be shown, by deploying optical switching technologies, the costly optical transceivers can be eliminated resulting in a cost-effective solution for DCNs.

5. Conclusions

LIGHTNESS network based on the innovative AoD multicluster approach provides an effective way to build large data centers in terms of performance and economic cost. In the first case, AoD multiclusters could scale data centers to several thousand servers providing full non-blocking network topology. It is shown that LIGHTNESS network will be effective to build large capacity Data Centers that handles the execution of multiple applications at the same time. It has been shown that these applications could be running on different racks in the system. There are racks that could be connected only through OCS and other racks that could use either OPS or OCS depending on the application need. The amount of racks in each of these groups depends on the size of the OCS. A mathematical model is provided to quantify the each of these group sizes. Applications could be mapped to each of these groups and take advantage of the benefits of both OCS and OPS. AoD-based data centers are configurable and flexible enough to be tailored to application needs. Long live traffic could use the OCS racks and short live traffic could be mapped on OPS racks.

It has been investigated the scalability limitations of this network architecture. It has been found that the scalability is mostly dependent on the size of the OCS. Larger OCS that provides a large number of ports could support larger OPS and larger number of clusters as expected. However, due to the non-linear resource requirements of OPS the larger OCS does not imply to support a larger number of servers in a cluster. There are certain sizes of OCS that actually support a larger number of servers. A mathematical model is provided to find this optimal point

Additionally, it has been investigated two approaches to scale this network to larger number of servers. The first approach relies on using smaller OPS. This technique could boost the number of racks in an AoD cluster without reducing the number of total OPS ports supported. The other technique is based on increasing the number of servers per rack using fast time division multiplexing techniques. This technique could increase by a factor of 8X times the number of servers supported without impacting significantly to HPC applications.

And finally, our study about the economic cost of LIGHTNESS network shows that LIGHTNESS is more cost effective solution than current data center networks based on electrical switches. The reason for that is that LIGHTNESS reduces significantly the number of expensive transceivers that current network are requiring. The dominant cost of LIGHTNESS is not coming from transceivers but from the optical switches.

References

[1] U. Hoelzle y L. A. Barroso, The Datacenter As a Computer: An Introduction to the Design of Warehouse-Scale Machines, 1st ed., Morgan and Claypool Publishers, 2009.

[2] S. Peng, B. Guo, C. Jackson, R. Nejabati, F. Agraz, S. Spadaro, G. Bernini, N. Ciulli y D. Simeonidou, «Multi-Tenant Software-Defined Hybrid Optical Switched Data Centre,» *J. Lightwave Technol.,* vol. 33, nº 15, pp. 3224-3233, Aug 2015.

[3] J. Titus, «DWDM Communications Relay on Basic Test Techniques,» EDN Network, 2000.

[4] Polatis, *Polatis Series 6000n Protection Services Switch,* Polatis.com, Ed., 2014.

[5] S. D. Lucente, J. Luo, R. P. Centelles, A. Rohit, S. Zou, K. A. Williams, H. J. S. Dorren y N. Calabretta, «Numerical and experimental study of a high port-density WDM optical packet switch architecture for data centers,» *Optical Express,* vol. 21, nº 1, pp. 263-269, 2013.

[6] W. Miao, S. D. Lucente, J. Luo, H. Dorren y N. Calabretta, «Low latency and efficient optical flow control for intra data center networks,» de *Proceedings of the European Conference and Exhibition on Optical Communication, Optical Society of America*, 2013.

[7] M. Technologies, *OMNeT++ InfiniBand Flit Level Simulation Model,* 2015.

[8] 2015. [En línea]. Available: http://www.top500.org/lists/2015/06/.

[9] M. A. Heroux., *Mantevo Home Page,* 2008.

[10] D. H. Bailey, E. Barszcz, J. T. Barton, D. S. Browning, R. L. Carter, L. Dagum, R. A. Fatoohi, P. O. Frederickson, T. A. Lasinski, R. S. Schreiber, H. D. Simon, V. Venkatakrishnan y S. K. Weeratunga, «The NAS Parallel Benchmarks; Summary and Preliminary Results,» de *Proceedings of the 1991 ACM/IEEE Conference on Supercomputing*, New York, NY, USA, 1991.

[11] M. a. E. Alizadeh, «On the Data Path Performance of Leaf-Spline Datacenter Fabrics,» de *High Performance Interconnects (HOTI), 2013 IEEE 21st Annual Symposium*, 2013.

[12] A. Andreyev, «Introducing data center fabric: the next-generation Facebook data center network».

[13] 2015. [En línea]. Available: https://www.broadcom.com/products/Switching/Data-Center/BCM56850-Series.

[14] D. C. -. M. S. v1.0., «OPEN Compute Project».

[15] 2015. [En línea]. Available: www.elpeus.com/categories.

[16] 2015. [En línea]. Available: www.industrialnetworking.com/Category.

Acronyms

ACK	Acknowledgement
AoD	Architecture on Demand
AWG	Arrayed Waveguide Grating
CPU	Central Processing Unit
DC	Data Centre
DCN	Data Centre Network
DWDM	Dense Wavelength Division Multiplexing
FPGA	Field Programmable Gate Array
FBG	Fiber Bragg Grating
HPC	High Performance Computing
IB	Infiniband
MPI	Message Passing Interface
MUX	Multiplexer
NACK	Negative Acknowledgement
NIC	Network Interface Card
OCS	Optical Circuit Switching
OF	OpenFlow
E/O	Electrical-Optical
OPS	Optical Packet Switching
QoS	Quality of Service
SDN	Software Defined Networking
SOA	Semiconductor Optical Amplifiers
TDM	Time Division Multiplexing
ToR	Top of the Rack