

A Comparative Survey on Optical Burst Switched Network Simulators

Terrance Frederick Fernandez, T. Megala, N.Sreenath

Department of Computer Science & Engineering, Pondicherry Engineering College, Puducherry – 605 014, INDIA
frederick@pec.edu, megala@pec.edu, nsreenath@pec.edu

Abstract—Optical Burst Switching (OBS) is the future of optical internet which provides a promising architecture to efficiently utilize WDM and to fulfill today's Internet traffic demands. Since this area is yet to mature there is a darn need to study its various protocols. Network simulator is an important tool for researchers and academicians to simulate and model the actual network at a reduced cost compared to the real test-bed experimentation. In this paper various OBS simulators are comparatively surveyed, while highlighting their merits and demerits.

Index Terms—Optical Burst Switching (OBS), All-Optical Networks, Network Simulator Tools.

I. INTRODUCTION

Internet, a boon to Science and humanity, is indispensable to our modern lives. Daybreaks in most houses checking emails, downloading music and listening to online radios. Driven by these human needs, the Internet grows exponentially and doubles its traffic every nine months before 1997 and every six months after that. According to [1], less than one in seven Americans were online in 1995. Today, two out of every three Americans are surfing the Web, exchanging e-mail, reading bank statements, checking ball scores and weather. This demand leads to a phenomenon in network called as traffic doubling [2]. The year 1990, hypertext and its supporting protocols made the World Wide Web, accessible to the public users. The World Wide Web transformed people's lifestyle like they can play travel agent and book all the elements of a vacation online, they can arrange for their bills to be paid automatically on absence or they can put a hold on mail delivery, find directions to tourist attractions and get a long-term weather forecast before leaving. Introduction of optical fiber cables decreased the cost of transmission bandwidth and thus made Internet affordable to laymen. Optical fiber allows multiplexing many signals at different centre wavelengths onto a single fiber. This is called "Wavelength Division Multiplexing (WDM)".

A. OBS Background

OBS offers a hybrid switching architecture that resembles both wavelength routing and packet switching. An OBS network consists of Electronic Edge Nodes (EN) and Optical Core Nodes (CN). The input EN is called as

Ingress and the output EN is called as Egress. Data is broken into variable sized entities called bursts, which is defined as the Digitized talk spurt or the data message that has intermediate granularity between a packet and amount of data in a circuit. A Burst consists of the payload called as Data Burst (DB) and a Burst Header Packet (BHP). DB is switched all-optically, while BHP undergoes O/E/O conversion at each hop. BHP precedes DB and is sent on a different wavelength with an "Offset Time (OT)". This OT allows the processing of the control packet and configuring switches until the arrival of data [3].

Buffers are absent at core nodes, headers can be processed at slower speeds, synchronization requirements are relaxed in OBS. OCS is suitable for constant rate traffic (voice traffic) but unsuitable for dynamic traffic. Faster header processing and strict synchronization are required in OPS due to lack of optical buffers. Only company that offers commercial OBS products is Matisse networks as the technology is still immature [4] and thus good for research. OBS can cut-through optical switches i.e., the burst may be transmitted before it is completely assembled or disassembled at intermediate nodes. The switching speeds of electronics cannot keep up with the transmission capacity offered by optics. Also optical switches handle large number of switching ports when compared with electronic switches.

B. Burst Contention and Losses

Burst Contention is a phenomenon that's common in OBS when two or more incoming Control Bursts compete for the same output wavelength at the same link and at the same time [4]. When contentions occur between bursts, one of them successfully reserves the channel and the remaining drops. This phenomenon is called as Burst Dropping and thus losses occur. This happens due to the absence of Optical buffers at the OBS Network as the DBs traverse an all-optical path.

C. Burst Assembly

Data from various traffic sources are assembled into Bursts of varied sizes. These are assembled at the Ingress before they are sent to the network. This assembly is made based on some algorithms and they are:

- Timer based Mechanism.
- Burst length (threshold) based Mechanism.
- Mixed Burst Assembly Mechanism.

- Optimized/Adaptive Burst Assembly Mechanism.

A burst is created and sent into ON at periodic time intervals and Bursts are of variable lengths for Timer based Burst Assembly Mechanism [5]. The burst lengths are fixed and are generated at non-periodic time intervals for Burst length based Burst Assembly Mechanism [5]. Combining packets of different classes to same bursts and packets are placed from head of the burst to the tail of the bursts in order of decreasing classes. This is called as composite burst assembly algorithm. In mixed burst assembly mechanism, the bursts are assembled and sent either if the timer expires or the burst length is reached. Dynamic adaptive threshold on burst length is set in order to optimize the overall performance in OBS for QoS sensitive traffic.

D. Burst Scheduling

Scheduling in OBS is essential for efficient use of channels/wavelengths and they can be Horizon or without void filling and without void filling:

- First Fit Unscheduled Channel (FFUC) Horizon scheduling
- Latest Available Unscheduled Channel (LAUC) Horizon scheduling
- First Fit Unscheduled Channel with Void Filling (FFUCVF)
- Latest Available Unscheduled Channel with Void Filling (LAUC-VF)
- Minimum End Void (Min-EV)

The latest time at which a channel is scheduled to use is “horizon”. With the above, the horizon scheduler selects the channel with latest horizon from the set of channels whose horizons are less than burst arrival times. The time complexity here is $O(\log h)$ [6]. VF keeps track of all voids in a channel and schedule bursts in one of the voids. If more than one void can fit a burst then the one with latest beginning time is assigned. The link utilization is higher than horizon scheduling but gets slower with huge number of voids [7]. MSV uses a geometric approach (binary search tree) that minimizes the distance between the starting time of the void and starting time of the burst.

This reminder of this paper is organized as follows. Section II describes an OBS Network model. In Section III, various Network simulators used on TCP over OBS connections are discussed with their merits and demerits. In Section IV, various simulators are compared. Section V deals with some OBS Test-bed implementations and the paper is concluded in Section VI.

II. OBS NETWORK MODEL

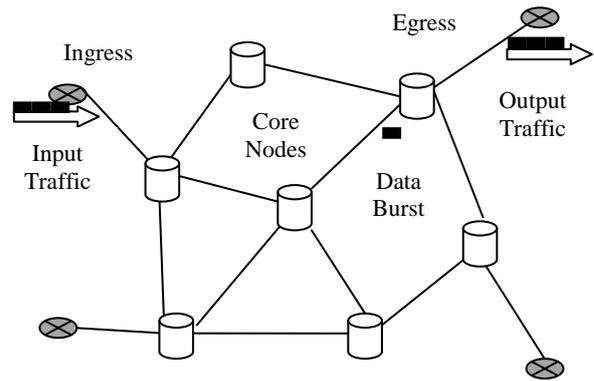


Fig 1: TCP over OBS Network

Fig. 1 shows OBS network architecture that consists of Ingress, Egress and core nodes connected by WDM links. The input traffic from multiple senders is assembled at the source edge node and is transmitted as bursts through high-capacity WDM links over the optical core. A destination edge node, upon receiving a burst, disassembles and delivers the data packets to the corresponding receivers. Source edge nodes are called as ingress and destination edge nodes are called as egress.

In OBS networks, data is transmitted as Data Bursts. Each burst consists of multiple packets (IP packets for an IP over WDM network or ATM packet), and a Burst Header Packet (BHP) is generated for each payload/data Burst. BHP contains the routing information to be used by core routers, the burst Offset Time (OT), the data burst length, the data wavelength carrying the burst, and the QoS information. A burst and its BHP are transmitted separately on different channels. In a burst transmission scenario, a BHP is sent out ahead of the burst by an OT in order to reserve resources at each intermediate core node along the path of the burst. While the BHP is processed at each intermediate node, the burst cuts through the pre-configured nodes all-optically. The OT ensures that, at each intermediate node, the BHP is processed prior to the burst arrival.

III. OBS NETWORK SIMULATORS

A. NCTUns

NCTU network simulator (National Chiao Tung University network simulator) is a module based network simulator developed at the Network and System Laboratory at National Chiao Tung University in 2002 [8]. Since it has module based structure, it supports real network applications. SimReal Incorporative is a virtual company that promotes the use of the NCTUns network simulator and emulator used for Network planning, research, application program, performance evaluation and educational needs, since its inception. This simulator deals mainly the OSI Data link and network layer protocols such as routing and switching. In the OSI

physical layer, it simulates the Bit Error Rate (BER), down time periods, bandwidth and propagation delay. For All-Optical Networks, it caters two simulation environments: the first one is a traditional OCS network, while the other one is the OBS network. It can support both wired and wireless communication links. An environment implemented in C++, it is an event driven simulator and is commercially licensed. NCTUns 4.0 (latest version) was tested on Red-Hat Fedora 7 Linux OS with kernel version 2.6.21. This simulator is used in [9], [10] and [11].

The main disadvantage here is the absence of commercial license. Also only few TCP implementations are dealt here that makes it unfeasible for simulation of every TCP implementation over OBS networks.

B. OBS Simulator

At North Carolina State University, USA, this network simulator was developed by Tend and Rouskas to model and simulate OBS networks, using C++. This network tool is unpublished; however there are published results of this simulator in [8]. Based on the Burst Loss Probability (BLP), simulation of different resource reservation protocols is implemented to study the performance of OBS, including network topologies. It defines the topology, scheduling algorithm, the number of edge nodes per core, the percentage of edge nodes that can generate bursts per core node and the propagation delay between core nodes. On command line, the program is called to give option parameters about the reservation protocols such as (JIT, JET, Horizon, JIT+ and JumpStart), the inter-arrival process (time between events), the burst generation ratio per node, the number of available data channels per link, the number of wavelength converters in each node, the time to process the setup message, the time to configure the optical cross-connect, and the propagation delay between edge and core nodes. Simulation results are only presented on console so advanced data output analysis by the means of plotting graphs is not available. The same happens with animation or real-time viewing, which are not supported. The main disadvantage is that it has no license and not available for download and use. Other disadvantages includes the unavailability of GUIs, usage of a fixed file name "n1.net"(created by its developers) with a particular semantics and syntax to run the any simulation and no Network animation output view.

C. OPNET

Optimized Network Engineering Tools (OPNET) was founded by OPNET technologies inc. (NASDAQ:OPNT). It could model and simulate simple, single core node model and multicore nodes. It is implemented using Object oriented programming and is a discrete event simulator. The most vital feature here is that it uses GUI based debugging and analysis and it does cater 32 and 64-bit fully parallel simulation. Also, OPNET accelerates the R&D process for analyzing and designing network also catering optical buffering (implementation of FDLs),

wavelength conversion and different assembly algorithms. This simulator is used in [12], [13] and [14].

The main disadvantage is that it supports only JET signaling protocol, thus JIT and Horizon are unsupported. The commercial licenses are available but the product is not free. Fig.2 shows the simulation output of OPNET.

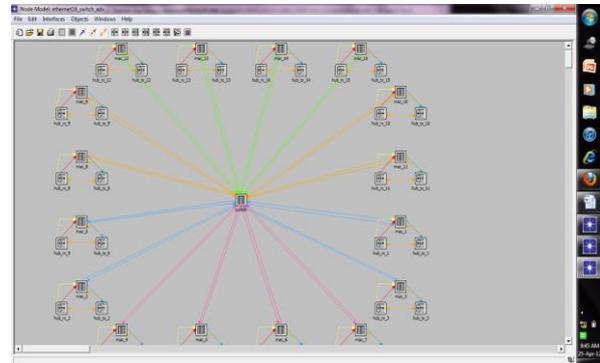


Fig 2: Campus Network Simulation on OPNET

D. JAVOBS

JAVOBS is developed by Oscar Pedrola from Technical University of Catalonia, USA. It is a Java based Simulator exclusive to OBS network simulation that uses JAVANCO framework [15]. Every OBS signaling protocols (such as JET, JIT, Horizon, E-JIT, JIT+) can be modeled here using hybrid discrete event modeling. Since this simulator is JAVA made, which is flexible, the complexity of building the simulator is reduced. Both C-OBS and E-OBS architectures can be implemented here [16]. This simulator is used in [17] for comparison of reservation protocols performance in C-OBS and E-OBS.

The main disadvantage is that runtime performance of JAVOBS is worse than those simulators implemented by C++ because of time consumed for the temporary compilation of user code to byte code. This code does not become executable code until the program is actually run.

E. OBS-ns Simulator

The OBS-ns simulator was the product of DAWN Networking Research Lab from University of Maryland. A redesigned version of DAWN Lab's simulator is the OIRC OBS-ns Simulator [18], a simulation tool developed by Optical Internet Research Center and Samsung Advanced Institute of Technology. It is a redesigned version of OWns, which uses an older version of ns-2. It aims to solve the problems of version 0.9 and improve the software, thus introducing new features. As its predecessor, OIRC OBS-ns is an event-driven simulator built on the ns-2, so it inherits its properties like the fact that object models are also developed in C++ and OTcl is still used for specifying and configuring the network simulation environment. Here it is necessary to build a routing table using shortest path routing and specify traffic streams as done in ns-2. There is also an extended list of OTcl parameters with default values that

is to be changed to specify details about OBS, such as the delay used of in FDL at end of link, the size of BHP and DB overhead, the burst timeout, the maximum burst size, the offset time for class and the edge node electronic buffer size. This simulator is modified and used in [19].

The main disadvantage here is lack of documentation about resource reservation protocols supported in this simulator. Furthermore, OBS-ns simulator implements only shortest path routing.

F. ADOBS

Ad-hoc event-driven simulator implemented in C++ language is an event driven simulator for OBS networks. The main motive behind creation of such a simulator is to study the performance of OBS Network layer. Protocols such as JET, Horizon can be simulated using this tool [16].

The disadvantages include complexity cost that is used to achieve speed and efficiency. All the concepts and operations are entirely dependent on the system's hardware. The input file is strictly predefined and not scripted unlike other network simulators.

G. DESMO-J

Developed in 1989 at the University of Hamburg, in the context of student's projects, DESMO-J (Discrete-Event Simulation and MODelling in Java) is an object-oriented framework can be used for modeling OBS. This Java implementation supports the discrete-event simulation i.e., all system state changes are supposed to happen at discrete points in time like queuing networks.

The DESMO was done in Modula-2, was an inspiration of DEMOS, a package for discrete-event simulation in Simula that was earlier developed by Birtwistle in 1979. In the year 1999, the core DESMO-J framework was completed. It was then extended in various aspects, like providing special components for the simulation of production systems or harbor logistics. This simulator is used in [20].

H. Owns

Owns (Optical Wavelength division multiplexing network simulator) was created as an extension to ns. It is honored as the first ns2 based OBS network simulator. The OIRC and OBS-NS are redesigned versions of OWNs. Optical switching node, multi-wavelength links, routing can be simulated using this one. It was developed by studying other simulators such as REAL, OPNET, NS and BONES. The NAM Output for this simulator is called as OWNAM.

The vital demerit here is the possibility of implementation of very limited number of scheduling, assembly and routing algorithm for OBS networks. To counter these demerits later was OIRC OBS-ns and nOBS were created based on ns2.

I. OBSIM

OBSim (Optical Burst Switch Simulator) is a event driven network simulator and the events are messages

[21]. These messages are generated by either by user or by network node. This simulator is used in [21].

OBSim make use of Java and is designed to implement OBS networks in form of objects [21]. It further allows to assess and compare the performance of signaling protocols and load profiles to a given network topology. This simulator is used to measure the performance of an OBS networks and a tool to predict the behavior of OBS network.

J. nOBS

Unlike Owns and OIRC OBS-ns, where only shortest path routing can be simulated, here in nOBS, any routing can be implemented using source routing [22]. Here, the Wavelength Converters and FDLs are combined into pools and are shared among all ports called as Share-per-Node architecture.

The nodes and links are modified into optical nodes and optical links. The address classifier is replaced by another one that differentiates TCP/IP packets from Bursts. This work is created by Guray Gurel, Onur Alprazolam and Ezhan Karasan at Bilken University, Turkey and supported by Scientific and Tech. Research Council of Turkey (TUBITAK). The simulation output for nOBS is given in Fig.2 and Fig.3.

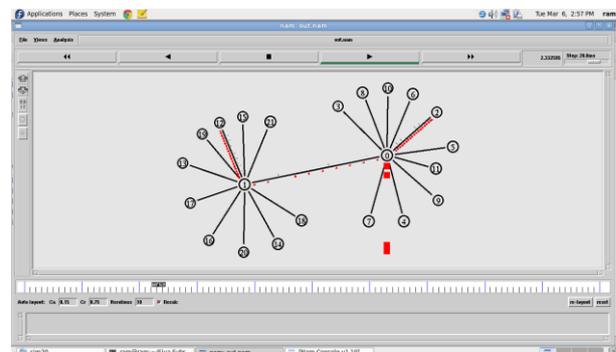


Fig 3: OBS Network with dumbbell topology represented on nOBS

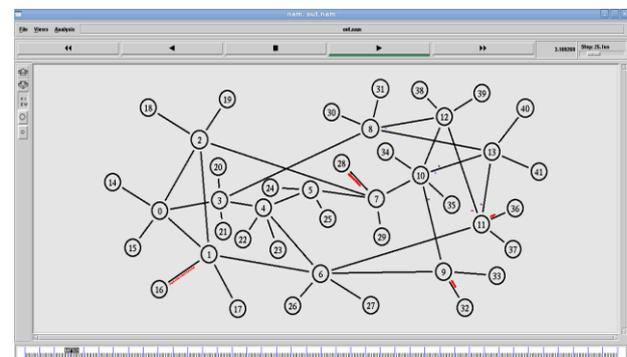


Fig 4: TCP/OBS Network in NSF topology represented on nOBS

K. OMNeT++

The primary function of OMNeT++ is not a network simulator, but a generic discrete event simulator framework to create scenarios, from the hard disk operation to the behavior of an Ethernet network.

The important advantage of OMNeT++ is that it is open source and has a Academic Public License which makes it free for non-commercial use [23]. Availability on all common platforms, like Windows, Mac OSX and Linux adds another advantage. All source code is in C++ and can be compiled with Microsoft Visual C++.

Apart from the advantages discussed above, this simulator suffers from few disadvantages. All OBS routing protocols cannot be modeled here.

L. IKRSimLib

The IKRSimLib is an object-oriented class library for event-driven simulation. It is implemented in C++ language. IKR Simulation Library V 2.5 is documented in [24]. First and foremost it was designed towards powerful support of performance evaluation of communication networks. However, as it was flexible so been applied to assess software system performance of statistically evaluate traffic measurement traces. This simulator is employed in [25], to simulate the Fiber Delay Lines.

IKRSimLib contains components like queue to servers. Added packages containing a TCP implementation like Reno, Vegas etc., and packet scheduler models were developed on top of the library. The merits of this simulator include simulation of asynchronous, non-periodic data transmission. Other benefits are GUI, Object-oriented design of simulation entities and easy extensibility. Again all OBS protocols cannot be modeled here, that is a vital demerit.

IV. SIMULATORS COMPARATIVE STUDY

This paper surveys twelve network simulators of TCP over Optical Burst Switched Networks. Among these eleven six are compared with one another. The comparison of OBS Simulators is done in TABLE 1 and TABLE 2.

TABLE 1 does the comparison on OBS signaling protocols, Programming language used and the license type. TABLE 2 does the comparison on Input and the output nature and the User Interfaces. This survey could be helpful to spot the best network simulator for a particular scenario.

Table 1. COMPARISON OF OBS SIMULATORS

Simulators	OBS Protocols	Programming Language	License Type
NCTUns	JET	C++	Commercial
OBS-ns		C++	Free Use
OBSsim	JET,JIT,JIT+ Horizon, Jump start	C++	Private Use
ADOBS	JET, Horizon	C++	Private Use
JAVOBS	JET,JIT,EJIT, JIT+ Horizon	JAVA	Private Use
nOBS	JET	C++	Free Use

Table 2. COMPARISON OF OBS SIMULATORS

Simulators	Input type	Output type	GUI
NCTUns	Graphical Model Construction	Plot Graph, Animation	Powerful GUI
OBS-ns	OTCL Script	Plot Graph	network animator
OBSsimulator	Script	Console output	NO GUI
ADOBS	Predefined input	Plot Graph	GUI
JAVOBS	Graphical, script, XML	Plot Graph	Powerful GUI
nOBS	Script	X Graph, nam	network animator

In addition to the simulator discussed above, an adaptive threshold burst assembly is simulated in [26] using Visual C++.

V. OBS TESTBED IMPLEMENTATION

Some of the works in OBS were experimented in real test fields called as a OBS Testbed which was given in [27]. Multi-QoS Traffic Transmission Experiments were carried out through testbed implementation in [28]. TCP performance on OBS was experimented in [29].

In [30], OBS network performance was tested in field trials on the Japan Gigabit Network II (JGN-II) testbed. Fast congestion control through the use of pre-calculated detour routes was also demonstrated. Demonstration of service aware bandwidth reservation in a multi-granular OBS test-bed was done in [31]. Demonstration of Novel Multi-Granular Switch Architecture on an Application-Aware End-to-End Multi-Bit Rate OBS was done in [32].

VI. CONCLUSION

Optical Burst Switched architecture is considered an effective one as core networking for large data requests and those applications that are delay sensitive. Since the OBS network is yet to mature it is mandatory to evaluate the performance. Also the OBS testbed could not be shipped to many countries including southern parts of Asia thus giving researchers and academicians that network simulators are the sole hope to analyze, design or implement OBS networks.

In this paper, survey on network simulators like NCTUns, OBSns, nOBS, DESMO-J, JAVOBS, OBSIM, OBS simulator, OWns, ADOBS, OMNeT++, OBS-ns and IKRSimLib that are used in scientific and academic research are done.

All that existing simulators except nOBS are not appropriate 'if and only if' a TCP over OBS Multicast network is considered to parameterize the Burst throughput, because they are either based on packet traffic instead of bursts, or do not cater every reservation protocols and parameters that can influence the TCP over OBS performance. Also when the learning curve is

considered, nOBS is easier when compared to others because of the share-per-node architecture and all the protocols are handled by agents.

REFERENCES

- [1] Pew Internet & American Life Project, <http://pewinternet.org/>, July 2012.
- [2] L. G. Roberts, "Beyond Moore's Law: Internet growth trends", *Journal of Packet Com Inc*, vol. 33, no 1, pp. 117-119, Jan. 2000.
- [3] Pushpendra Kumar Chandra, Ashok Kumar Turuk and Bibhudatta Sahoo, "Survey on Optical Burst Switching in WDM Networks", *International Conference on Industrial and Information Systems*, pp 83-88, December 2009.
- [4] Onur Ozturk, Ezhan Karasan, Member, IEEE, and Nail Akar, "Performance Evaluation of Slotted Optical Burst Switching Systems with Quality of Service Differentiation", *Journal of lightwave technology*, vol. 27, no. 14, July 15, 2009.
- [5] Farid Farahmand, Jason Jue, Vinod Vokkarane, "A Layered Architecture for Supporting Optical Burst Switching", *Proc. Advanced Industrial Conference on Telecommunications*, pp. 213-218, October 17, 2005, Dallas, Texas, USA.
- [6] Georgios I. Papadimitriou, Chrisoula Papazoglou and Andreas S. Pomportsis, "Optical Switching: Switch Fabrics, Techniques, and Architectures", *Journal of lightwave technology*, Vol.21, No.2, pp. 384-405, February 2003.
- [7] Yong Liu, Gurusamy Mohan, Kee Chaing Chua and Jia Lu, " Multipath Traffic Engineering in WDM Optical Burst Switching Networks" , *IEEE Transactions on Communications*, Vol. 57, No. 4, pp. 1099 – 1108, April 2009.
- [8] Vasco N.G.J.Soare, Iúri D.C. Veiga and Joel J. P. C. Rodrigues, "OBS Simulation Tools : A Comparative Study", 2009.
- [9] S.Y. Wang, "Using TCP Congestion Control to Improve the Performances of Optical Burst Switched Networks", *IEEE Conferences*, 2003, Hsinchu, Taiwan.
- [10] Mohamad Abdulatef Al-Shargabi and Abdulwahab Abid, "The Impact of OBS Burst Aggregation on VBR Performance", *Proceedings of the 2007 IEEE International Conference on Telecommunications and Communications*, 14-17 May 2007, Penang, Malaysia.
- [11] Alexios Louridas, Kalliopi Panagiotidou and Nathan J. Gomes, "Simulation of Optical Burst Switching Protocol and Physical Layers", 2002.
- [12] Óscar González, Ignacio de Miguel, Noemí Merayo et.al., "The impact of delayed ACK in TCP flows in OBS networks", *Innovation Strategy, Telefonica*, 2004.
- [13] Yi Xu, "Signaling protocol to reduce blocking probability in Optical Burst Switching mesh networks with limited wavelength conversion capabilities ", *Third International Conference in communications and Networking*, pp 448-453, 2008, China .
- [14] Gipsper .T, "An all-optical network architecture", *Journal of Lightwave Technology*, vol 14 , no 5, pp 693-702, May 1996.
- [15] Oscar Pedrola, Sébastien Rumley and Miroslaw Klinkowski, "Flexible Simulators for OBS Network Architectures", *International Conference on Transparent Optical Networks (ICTON)*, pp. 117-122, 2008.
- [16] Mohammed.A.Al-Shargabi, Abdul Samad Ismail and Sevia M.Idrus, "A Feature Comparative study of Real-Time Traffic Simulation over OBS Network", *International Journal of Computer Applications*, vol.21,no.4, pp. 13-16, May 2011.
- [17] Joan Triay and Cristina Cervell ó-Pastor "Performance Comparison of Reservation Protocols of COBS and EOBS", *International Conference on Transparent Optical Networks (ICTON)*, pp. 17-22, 2008.
- [18] Tran Minh Trung, "Optical Internet Research Center OIRC OBS-ns Simulator", *October 2007*.
- [19] Orawiwattanakul, "Fair Bandwidth Allocation in Optical Burst Switching Networks", *Journal of Lightwave Technology*, vol 27, no 16, pp 3370 - 3380, 15th Aug 2009.
- [20] Joan Triay and Cristina Cervell ó-Pastor, "QoS Multi-Spanning Tree for Optical Burst Switching", *IT News* 2008.
- [21] Joel J. P. C. Rodrigues, Nuno M. Garcia and Pascal Lorenz, "Object-oriented modelling and simulation of Optical Burst Switching Networks", *IEEE Communication Society, GLOBECOM*, pp 288-292, 2004, Portugal.
- [22] Guray Gurel , Onur Alparslan and Ezhan Karasan, "nOBS: an ns2 based simulation tool for performance evaluation of TCP traffic in OBS networks", 2007.
- [23] Felix Espina, Javier Armendariz, Naiara Garc á et.al, "OBS network model for OMNeT++: A performance evaluation", *ICST 2010, March 15–19, Torremolinos, Malaga, Spain*.
- [24] Institute of Communication Networks and Computer engineering, University of Stuttgart, "The IKR Simulation Library", www.ind.unistuttgart.de/IKRSimLib, June 2004..
- [25] Maria Cristina Fischer de Toledo, Dr. Wagner L. Zucchi, "Simulation of a optical burst switch using fiber delay lines", *IEEE* 2005.
- [26] Burak Kantarci, Sema Oktug, "Adaptive Threshold Based Burst Assembly In OBS Networks", *IEEE CCECE/CCGEI, Ottawa, May 2006*.
- [27] Guo H., "A Testbed for Optical Burst Switching Network," *OFC 2005, March 2005.OFC2005*.
- [28] Guo H., "Multi-QoS Traffic Transmission Experiments on OBS Network Testbed," *ECOC'05*.September 2005.
- [29] Zhang W., Wu J., Xu K. and Lin J.T, "TCP performance experiment on OBS network testbed", *IEEE*.
- [30] Akio Sahara, Member, Takashi Ono, Jun Yamawaku, Atsushi Takada et.al., "Congestion-Controlled Optical Burst Switching Network with Connection Guarantee: Design and Demonstration", *Journal of Lightwave Technology*, Vol. 26, No. 14, July 15, 2008.
- [31] G. Zervas, Y. Qin, R. Nejabati, D. Simeonidou, "Service Oriented Optical Burst Switched Edge and Core Routers for Future Internet", *IEEE* .
- [32] Georgios Zervas, Lida Sadeghioon, Dimitrios Klondis, et.al., "Demonstration of Novel Multi-Granular Switch Architecture on an Application-Aware End-to-End Multi-Bit Rate OBS Network Testbed", *33rd European Conference on Optical Communication (ECOC 2007)*, Berlin, Germany, September 2007.



Terrance Frederick Fernandez received his M.Tech Degree in Distributed Computing Systems from Pondicherry Engineering College, India, in 2008 and B.Tech Degree in Computer Science and Engineering from Sri Manakula Vinayagar Engineering College (affiliated to Pondicherry University), India, in 2006.

He is currently doing research at the Information Security Laboratory, Department of Computer Science and Engineering, Pondicherry Engineering College, India.



Asst. Prof. T.Megala received her M.Tech Degree in Distributed Computing Systems from Pondicherry Engineering College, India, in 2012, her M.C.A Degree from Sri Manakula Vinayagar Engineering College (affiliated to Pondicherry University) and her Bachelor degree in Computer Science from Pondicherry University.

She is currently working as an Assistant Professor at Sri Manakula Vinayagar Engineering college (affiliated to Pondicherry University), India.



Prof. Dr. N.Sreenath received his Ph.D Degree in Computer Science and Engineering from Indian Institute of Technology, Chennai, India, in 2003, his M.Tech Degree from University of Hyderabad, in 1990 and his B.Tech in Electronics and Communication Engineering from JNTU Hyderabad, India in 1987.

His area of expertise is WDM Optical Networks. He has been with the Department of Computer Science and engineering at the Pondicherry Engineering College, India, since 2003, where he is currently a Professor.

How to cite this paper: Terrance Frederick Fernandez, T. Megala, N.Sreenath, "A Comparative Survey on Optical Burst Switched Network Simulators", IJCNIS, vol.6, no.5, pp.35-41, 2014. DOI: 10.5815/ijcnis.2014.05.05