

Cross-layer hybrid and optical packet switching

Artur Minakhmetov

▶ **To cite this version:**

Artur Minakhmetov. Cross-layer hybrid and optical packet switching. Networking and Internet Architecture [cs.NI]. Institut Polytechnique de Paris, 2019. English. NNT : 2019IPPAT006 . tel-02481270

HAL Id: tel-02481270
https://pastel.hal.science/tel-02481270

Submitted on 17 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2019IPPAT006

Thèse de doctorat

Commutation de paquets optique et hybride multicouches

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom Paris

École doctorale n°626 École doctorale de l'Institut Polytechnique de Paris (ED IP Paris)
Spécialité de doctorat : Réseaux, Informations et Communications

Thèse présentée et soutenue à Paris, le 4 Décembre, par

ARTUR MINAKHMETOV

Composition du Jury :

Hind Castel Professeur, Télécom SudParis, France	Président
Nicola Calabretta Associate Professor, Technische Universiteit Eindhoven, Pays-Bas	Rapporteur
Stefano Secci Professeur, CNAM, France	Rapporteur
Mounia Lourdiane Maître de conférences, Télécom SudParis, France	Examinatrice
Emmanuel Lochin Professeur, ISAE-SUPAERO, France	Examinateur
Daniel Kilper Professeur, The University of Arizona, États-Unis	Examinateur
Nihel Djoher Benzaoui Ingénieure de recherche, Nokia Bell Labs, France	Invitée
Cédric Ware Maître de conférences – HDR, Télécom Paris, France	Directeur de thèse
Luigi Iannone Maître de conférences, Télécom Paris, France	Co-directeur de thèse

Abstract

Transparent optical telecommunication networks constitute a development step from all-electronic networks. Current data network technologies already actively employ optical fibers and transparent networks in the core, metro, and residential area networks. However, these networks still rely on Electronic Packet Switching (EPS) for packets routing, constituting obligatory for each packet optical-to-electronic-to-optical (OEO) signal conversion. On the other hand, Optical Packet Switching (OPS), seemed to be as replacement of EPS, has long promised performance and energy consumption improvements by going away from OEO conversions; however, the absence of practical optical buffers made OPS highly vulnerable to contention, incurring performance reduction, and getting in the way of profiting from OPS gains.

The subject of this research lies in the investigation of the performance of OPS networks under all-optical and hybrid switches, while server-side transmission activities are regulated by Transport Control Protocols based on Congestion Control Algorithms (TCP CCAs). We consider that OPS could be enabled by use hybrid switch, i.e. device-level solution, as well by use of specially designed TCP CCAs, i.e. network-level solution, giving birth to Hybrid Optical Packet Switching (HOPS) networks.

We extensively study OPS, HOPS and EPS types of Data Center Networks (DCN) coupled with different TCP CCAs use by following the next three axes of DCN performance: Throughput, Energy Consumption, and Latency. As for TCP CCAs we consider not only existing but also newly developed solutions. If Stop-And-Wait (SAW), Selective Acknowledgment (SACK), modified SACK (mSACK) and Data Center TCP (DCTCP) are already known to the world, Stop-And-Wait-Longer (SAWL) is newly presented and is designed to bring the best out of the HOPS DCN. As a result, it is shown that hybrid switch solutions significantly outperform bufferless all-optical switches and reach the level of all-electronic switches in DCNs in terms of throughput. In terms of energy consumption, hybrid solutions can save up to 4 times on energy on switching compared to all-electronic solutions. As well HOPS DCNs can exhibit microseconds-scale average latencies, surpassing EPS and performing on the level with OPS.

The question of the introduction of Classes of Service to HOPS DCN is also investigated: it was found that class-specific switching rules to hybrid switch can ameliorate the performance of certain classes without almost performance loss in others.

Keywords: Packet-switched networks, Packet Switching, Transparent networks, Optical Switches, Hybrid Switches, TCP, Congestion control, CCA, Data Center, Data Center Networks

Résumé

Les réseaux de télécommunication transparent constituent une étape de développement des réseaux entièrement électroniques. Les technologies de réseau de données actuelles utilisent déjà activement les fibres optiques et les réseaux transparents dans les réseaux centraux, métropolitains et résidentiels. Toutefois, ces réseaux reposent toujours sur la commutation électronique de paquets (EPS) pour le routage des paquets, qui rend obligatoire pour chaque paquet d'avoir une conversion de signal optique à électronique à optique (OEO). D'autre part, la commutation optique de paquets (OPS), qui semblait remplacer le système EPS, promet depuis longtemps des améliorations en termes de performances et de consommation d'énergie en s'éloignant des conversions OEO ; Cependant, l'absence de buffers optiques pratiques rendait OPS extrêmement vulnérable aux contentions, entraînant une réduction des performances et empêchant de tirer profit des gains de l'OPS.

L'objetif de cette recherche est d'étudier la performance des réseaux avec des commutateurs tout optiques et hybrides, tandis que les activités de transmission côté serveur sont régies par des protocoles de contrôle de transport basés sur des algorithmes de contrôle de congestion (TCP CCA). Nous considérons que l'opération OPS pourrait être activée en utilisant un commutateur hybride, c.à.d. une solution au niveau de l'appareil, ainsi que des TCP CCA spécialement conçus, c.à.d. une solution au niveau du réseau, donnant naissance à des réseaux hybrides à commutation de paquets optique (HOPS).

Nous étudions les réseaux de centres de données (DCN) de type OPS, HOPS et EPS associés à différentes TCP CCAs en suivant les trois axes de la performance : débit, consommation d'énergie et latence. En ce qui concerne les TCP CCA, nous considérons non seulement les solutions existantes, mais également celles développées. Si Stop-And-Wait (SAW), Selective Acknowledgment (SACK), SACK modifié (mSACK) et Data Center TCP (DCTCP) sont déjà connus, Stop-And-Wait-Longer (SAWL) est présenté ici et conçu pour tirer le meilleur du HOPS DCN. Il est démontré que les solutions de commutateurs hybrides surpassent de manière significative les commutateurs tout optiques sans buffer et atteignent le niveau de commutateurs tout électroniques en termes de débit du reseau. En termes de consommation d'énergie, les solutions hybrides peuvent économiser jusqu'à 4 fois plus d'énergie de la commutation par rapport aux solutions tout électroniques. De plus, les DCN HOPS peuvent atteindre des latences moyennes à l'échelle des microsecondes, dépassant ainsi les EPS et se situant au même niveau que les OPS.

La question de l'introduction de classes de service dans HOPS DCN est examinée : on constate que les règles de commutation spécifiques en commutation hybride peuvent améliorer la performance de certaines classes sans pertes significatives d'autres.

Mots clés : Réseaux à commutation de paquets, commutation de paquets, réseaux transparents, commutateurs optiques, commutateurs hybrides, TCP, contrôle de congestion, CCA, centre de données, réseaux de centres de données

To my mother Roza
To my beloved Nastia

Acknowledgments

I would like to thank the jury members, Nicola Calabretta, Stefano Secci, Hind Castel, Mounia Lourdiane, Emmanuel Lochin, Daniel Kilper, Nihel Djoher Benzaoui for reading through my thesis and asking relevant and interesting questions during my thesis defense.

I thank my thesis advisers, Cédric Ware, Luigi Iannone for advising me, directing me and working side-by-side with me through all of these years of my Ph.D. Thanks to you I had a very fruitful Ph.D. with 4 conference papers on the top international OFC conference, which I consider our common achievement, not speaking about our 2 journal papers. Thank you for supporting me, believing in me, providing me with all means to conduct my research. Special thanks for providing me with the opportunity to go to a research visit to Columbia University, which let me broaden my research domain and lead a fruitful collaboration with Prof. Gil Zussman group.

I thank the team from Columbia University: Craig, Tingjun and Gil Zussman. Thanks for accepting me for the visit, for working me side-by-side on the work on our project on handover of wireless traffic, which eventually led to a well-earned accepted paper to publication.

I feel enormous gratitude towards my family. Thank you to my two most important women in my life: my mother Roza and my partner Anastasia. Thanks to your love and support I was able to go through this challenging life of a Ph.D. student. Thank you for being there for me, listening to me and bearing with me. I appreciate your being in my life tremendously. Thanks to Dina, my cousin, for our special and warm relationship.

I thank my friends for their warm and deep friendship. Thanks to my friends from Russia: Anton Ch., Anton Sch., Vova Ts., Jaeyeol R., Grisha Ch. I appreciate a lot Anton's Ch. coming personally to my defense to support me from far away, the distance is not an obstacle to our friendship! Thanks to my friends and colleagues from Télécom Paris: Samet, Akram, Abby, Julien, Alaa, Vincent. Thank you for our warm and heated discussions, supporting each other on hard moments, our after-telecom evenings! Thanks to my friends from Paris: Dima, Sérioja, Anton, Marina B., Natasha D., Oksana, Daria, Tatiana, Sasha, Misha, Katia, Ferdinand, and many others! Thank you for our special moments together, for all our ups and downs we had.

Thanks to my colleagues from my department for creating a safe, comfortable and willing to go back working environment, thanks to Bruno Thedrez, Hamidou Kone, Chantal Cadiat and Yvonne Bansimba.

Thank you all!

Contents

Abstract ii

Résumé iii

Acknowledgments v

Contents vi

List of Figures viii

List of Tables xi

Glossary xii

1 Introduction 1
- 1.1 Introduction into Optical Networks . . . 1
- 1.2 Optical Networks for Data Centers . . . 8
- 1.3 Thesis Structure . . . 9
- 1.4 Thesis Contributions and List of Publications . . . 9

2 State of the Art on Hybrid Optical Packet Switching 11
- 2.1 Network Strategies for Contention Resolution . . . 12
- 2.2 Hybrid Optical Packet Switching . . . 13
- 2.3 Conclusion . . . 16

3 Research Framework 19
- 3.1 Optical Packet Switching Model . . . 19
- 3.2 Hybrid Optical Packet Switching Model . . . 20
- 3.3 Electronic Packet Switching Model . . . 22
- 3.4 TCP SAW – Reference CCA for OPS DCN . . . 23
- 3.5 Simulation of Communications in a Data Center Network . . . 24
- 3.6 Conclusion . . . 27

4 Congestion Control Algorithms and their Performance in DC Network 29
- 4.1 TCP Stop-And-Wait-Longer . . . 29
- 4.2 TCP SACK and TCP mSACK . . . 33
- 4.3 Buffer Size of a Hybrid Switch and Its Influence on Latency . . . 41
- 4.4 Conclusion . . . 44

5 Energy Consumption in DC Network on Switching **45**
5.1 On the Metric for Energy Consumption For Data Transport 46
5.2 Energy Consumption on Switching in HOPS Data Center 47
5.3 Conclusion 49

6 Latency in DC Network on Different Switching Mechanisms **51**
6.1 DCTCP basics 51
6.2 Study of DCTCP performance in HOPS and OPS DCN 53
6.3 Latencies of different CCAs achievable in DC 61
6.4 Conclusions 63

7 Hybrid and Optical Packet Switching Supporting Different Service Classes in DC Network **65**
7.1 Class Specific Switching Rules in OPS and HOPS 65
7.2 Study Conditions 67
7.3 Advantages of using Class Specific Switching Rules 69
7.4 Conclusions 70

8 Conclusions and Future Work **73**
8.1 Summary and Conclusions 73
8.2 Future Research Directions 74

A DCTCP parameters influence on Network Performance **75**

Bibliography **83**

List of Figures

1.1 Optical Network Segments 2
1.2 Open Systems Interconnection Model 3
1.3 Common Protocols Stack in Optical Networks 4
1.4 Optical Transport Network (OTN) Protocol Wrapper 5
1.5 Second Degree ROADM operation principle 5
1.6 Possible Evolution of Protocol Stack towards All-Optical Networks 6

2.1 Architecture of a hybrid optical packet switch 15

3.1 General architecture of all-optical packet switch 19
3.2 General architecture of hybrid optical packet switch 21
3.3 General architecture of all-electronic packet switch 22
3.4 TCP SAW working principle 23
3.5 File Size Distribution if to consider 1024 random files. 24
3.6 Fat-tree topology network, connecting 128 servers with 3 layers of switches. 26

4.1 Network throughput dependence on TCP SAWL parameter $p \in \{0,1,4\}$ and number of buffer I/O ports n_e for: a) $l_{link} = 10$ m, b) $l_{link} = 100$ m. 30
4.2 Network throughput dependence on TCP SAWL parameter $p \in \{4,5,6\}$ and number of buffer I/O ports n_e for: a) $l_{link} = 10$ m, b) $l_{link} = 100$ m. 31
4.3 Buffer Occupancy of hybrid switches ($n_e = 8$) of different levels to transmit same set of 1024 files arriving at 10^9 req./s and regulated by: left) by TCP SAW ($kp = 0$), right) by TCP SAWL ($p = 4$) 32
4.4 Example of Congestion Window evolution for TCP SACK. Dots represent CWND size that is evaluated each time an ACK is received or RTO times out. 34
4.5 Example of Congestion Window evolution for TCP mSACK. Dots represent CWND size that is evaluated each time an ACK is received or RTO times out. 35
4.6 Data Center or LAN network with $l_{link} = 10$ m throughput dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK 37
4.7 Data Center or LAN network with $l_{link} = 100$ m throughput dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK 38
4.8 Data Center or LAN network with $l_{link} = 1$ km throughput dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK 41

4.9 Data Center or LAN network with $l_{link} = 10$ km throughput dependence *in log-scale* on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK . . 42
4.10 Maximum Buffer Size occurred during transmission of batch of 1024 Random Files, averaged. Dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK 43
4.11 99th Percentile of RTT for the $l_l ink = 10$ m, of Hybrid OPS ($n_e = 8$) network, depended on TCP CCA 44

5.1 "Bit transport energy factor" for 1 packet in EPS network: $\frac{9064\times6+64\times6}{9000} = \mathbf{6.085}$ 46
5.2 "Bit transport energy factor" for 1 packet in OPS network: $\frac{9064\times1+64\times1}{9000} = \mathbf{1.014}$ 46
5.3 Network throughput dependence on TCP CCA and switch type: a) $l_{link} =$ 10 m, b) $l_{link} = 100$ m. 47
5.4 Transmission energy cost dependence on TCP CCA and switch type: a) $l_{link} =$ 10 m, b) $l_{link} = 100$ m. 47

6.1 Difference on instantaneous buffer size (queue length) with different TCP . . 52
6.2 DCTCP working principle 54
6.3 Example of CWND evolution on DCTCP, taken from [84] 54
6.4 Throughput dependence on DCTCP parameters: a) $k = 9064$ B, $m = 1$, b) $k = 27192$ B, $m = 1$ c) $k = 9064$ B, $m = 2$, d) $k = 27192$ B, $m = 2$ 56
6.5 99th percentile of RTT dependence on DCTCP parameters: a) $k = 9064$ B, $m = 1$, b) $k = 27192$ B, $m = 1$ c) $k = 9064$ B, $m = 2$, d) $k = 27192$ B, $m = 2$ . . 57
6.6 99th percentile of FCT dependence on DCTCP parameters: a) $k = 9064$ B, $m = 1$, b) $k = 27192$ B, $m = 1$ c) $k = 9064$ B, $m = 2$, d) $k = 27192$ B, $m = 2$ . . 58
6.7 Buffer Occupancy of switches of different levels to transmit same set of 1024 files arriving at 10^9 req./s: left) HOPS,$n_e = 8$ over TCP SACK ($k =$ 2719200 B,$m = 1$,$g = 0.00$), right) HOPS,$n_e = 8$ over DCTCP ($k = 27192$ B,$m =$ 1,$g = 0.06$) 60
6.8 Buffer Occupancy of switches of different levels to transmit same set of 1024 files arriving at 10^9 req./s: left) EPS,$n_e = 8$ over TCP SACK ($k = 2719200$ B,$m =$ 1,$g = 0.00$), right) EPS,$n_e = 8$ over DCTCP ($k = 27192$ B,$m = 1$,$g = 0.06$) 60
6.9 Throughput dependence on CCA and load 62
6.10 Average FCT in a DCN 62
6.11 99th percentile FCT in a DCN 62
6.12 Average RTT in a DCN 62
6.13 99th percentile RTT in a DCN 62

7.1 Example of Class-specific Routing Rules for hybrid switch with $n_e = 1$. 67
7.2 DC network's throughput for connections: a) Reliable (R) connections, b) Not-So-Fast (F̃) connections, c) Default (D) connections, d) Overall Network Performance 68
7.3 DC network's Flow Completion Time for connections: a) Reliable (R) connections, b) Not-So-Fast (F̃) connections, c) Default (D) connections, d) Overall Network Performance 69
7.4 Mean PLR of Reliable (R) Connections 71

A.1 DCN ($l_{link} = 10m$) Average Throughput dependence on DCTCP parameters 76

A.2 DCN ($l_{link} = 100m$) Average Throughput dependence on DCTCP parameters 77
A.3 DCN ($l_{link} = 10m$) RTT 99th Percentile dependence on DCTCP parameters . 78
A.4 DCN ($l_{link} = 100m$) RTT 99th Percentile dependence on DCTCP parameters 79
A.5 DCN ($l_{link} = 10m$) FCT 99th Percentile dependence on DCTCP parameters . 80
A.6 DCN ($l_{link} = 100m$) FCT 99th Percentile dependence on DCTCP parameters 81

List of Tables

2.1 Hybrid Switch Candidates for Hybrid Optical Packet Switching 14

4.1 Key differences between TCP SAW and TCP SAWL RTO calculation 30

Glossary

ACK	Acknowledgement
AIMD	Additive Increase Multiplicative Decrease
AO	All Optical
AVG	Average
AWG	Arrayed Waveguide Grating
BBU	Baseband Unit
C-RAN	Cloud Random Access Network
CBOSS	Cloud Burst Optical-Slot Switching
CCA	Congestion Control Algorithm
CDF	Cumulative Distribution Function
CE	Congestion Experienced
CoS	Class of Service
CPU	Central Processing Unit
CWND	Congestion Window
DC	Data Center
DCN	Data Center Network
DCTCP	Data Center Transport Control Protocol
DOS	Datacenter Optical Switch
DUP	Duplicate
DWDM	Dense Wavelength Division Mutliplexing
ECE	ECN Echo
ECN	Explicit Congestion Notification
EO	Electronic-Optical
EPS	Electronic Packet Switching
FCT	Flow Completion Time
FDL	Fiber Delay Lines
FIFO	First In First Out
FIN	Final
FPGA	Field Programmable Gate Array
FTP	File Transfer Protocol
GB	Giga Bytes
HOPR	Hybrid Optical Packet Router
HOPS	Hybrid Optical Packet Switching
HPC	High Performance Computer
IETF	Internet Engineering Task Force
IP	Internet Protocol
I/Q	In-Phase/Quadrature
I/O	Input/Output

L1	Level 1
L2	Level 2
L3	Level 3
LAN	Local Area Network
LIONS	Low-Latency Interconnect Optical Network Switch
MAC	Media Access Control
mAIMD	modified Additive Increase Multiplicative Decrease
MAN	Metropolitain Area Network
MB	Mega Bytes
MEMS	Microelectromechanical Systems
MPNACK	Multi-Hop Negative Acknowledgment
mSACK	modified Selective Acknowledgment
MSS	Maximum Segment Size
MTU	Maximum Transmission Unit
MZI	Mach-Zender Interferometer
NACK	Negative Acknowledgement
NIC	Network Inteface Controller
NSF	Not-So-Fast
OBS	Optical Burst Switching
OCS	Optical Circuit Switching
OE	Optical-Electronic
OEO	Optical-Electronic-Optical
OPS	Optical Packet Switching
OPU	Optical Payload Unit
OSI	Open Systems Interconnection
OTN	Optical Transport Network
PDF	Probability Disctribution Function
PDU	Protocol-specific Data Unit
PLR	Packet Loss Ratio
PON	Passive Optical Network
PS	Packet Switching
RAN	Random Access Network
ROADM	Reconfigurable Optical Add & Drop Multiplexer
RTO	Retrensmission Time Out
RTT	Round Trip Time
SACK	Selective Acknowledgement
SAW	Stop-And-Wait
SAWL	Stop-And-Wait-Longer
SDH	Synchronous Digital Hierarchy
SDN	Software Defined Network
SMSS	Sender Maximum Segment Size
SOA	Semiconductor Optical Amplifier
SONET	Synchronous Optical NETwork
SYN	Synchronize sequence
TCP	Transport Control Protocol
ToR	Top of Rack
TWC	Tunable Wavelength Converter
UDP	User Datagram Protocol

WAN	Wide Area Network
WC	Wavelength Converter
WDM	Wavelength Division Multiplexing

Chapter 1

Introduction

1.1 Introduction into Optical Networks

1.1.1 Optical Networks as a Driver and Enabler

In the modern world, telecommunications play a huge role in many aspects of our lives. People's work, entertainment, social life activities are all closely tied with telecommunications. Let's just take an example of an average work-day: in the morning we read news on phones and tablets online; during the day we plan meetings, create documents and spreadsheets online, manage people and projects online; in the evening we might go chat on our favorite social network online and interact with other people online; others may go out for a movie or restaurant off-line, but will select them online and soon after leave a thorough review as well online; we go asleep by checking our emails online. All these online or internet activities are enabled and exist solely thanks to the possibility of telecommunication.

Let us consider just a case of morning news read from your favorite newspaper web-site (actual paper is no longer needed). The tablet or phone sends and receives traffic through a wireless connection to the nearest base station, the base station needs a connection to the network that could route the traffic locally and internationally, to land this traffic into edge or cloud data center. The path of traffic doesn't stop there: the request for some special article is processed in the data center, which has numerous interconnected servers, which work hard to construct a reply in the form of data containing a web-page with the desired article. And only after all of these processes, the data is sent back to repeat the whole path of the intertwined network to be landed on the user's phone or tablet. To support such present day telecommunications, efficient networks have to be constructed. An efficient network has several requirements, the most important are: bandwidth, latency, connectivity, and cost.

Bandwidth requirement means that a network has to support a certain amount of traffic, which is measured in the amount of data (in bytes or bits) sent per amount of time. According to the latest Cisco Visual Networking Index [1] global IP traffic will reach 396 Exabytes (1 Exabyte equals 1 000 000 Terabytes) by 2022 per month, i.e. 3 times 2017's. This ever-increasing traffic passes through different segments of the network as shown in Fig. 1.1: users' traffic enters a network from multiple points through an access network then joining Metropolitan Area Network (MAN) and then Core Network. Such combining and multiplexing of users' traffic impose different requirements on bandwidths of MAN and Core Network. The bigger the network segment becomes, the

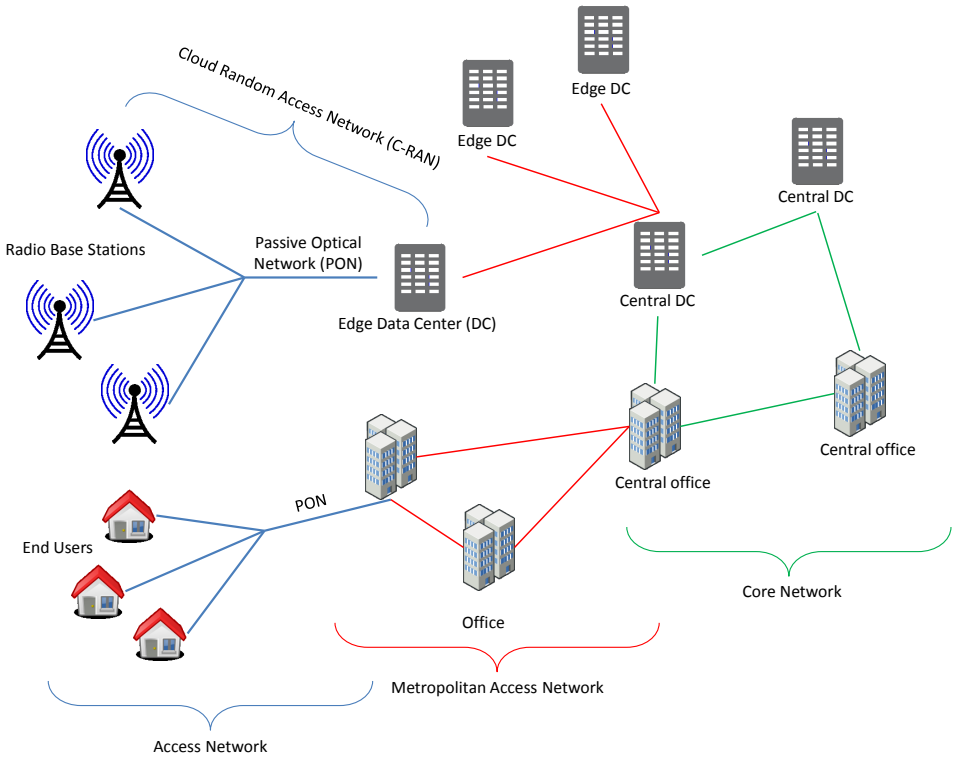


Figure 1.1 – Optical Network Segments

bigger traffic the network has to support. If it weren't for the enabling bandwidth of optical networks, this ever-increasing demand on traffic distributed over MAN and Core Network couldn't be answered. Optical networks can offer a wide variety of link speeds, that may meet the need of any segment of the network, offering high speeds going up to hundreds of Gigabits per second for short links [2] and even tens of Terabits per second over a single mode fiber [3] for transcontinental communication.

Latency, alongside with bandwidth, forms another requirement: if bandwidth defines the amount of traffic users exchange, latency defines a delay of such exchange of information. It is usually defined as a Round-Trip-Time (RTT) and measured in seconds, meaning time needed for a request sent by the user or server to be answered by server or user (respectively) in the form of a received response. Latency becomes a very important parameter in the future generation of mobile and data center networks. Latencies in the case of mobile networks: future 5G network standards require less than 1 ms of latency [4]. The answer to that requirement mostly consists of the application of optical networks [5], which are limited only by the speed of light. Such networks allow using Cloud Random Access Network (C-RAN), which means: 1) decoupling wireless signal processing from the base station and processing it in a cloud data center; 2) connecting multiple radio base stations by a Passive Optical Network (PON) [6]. Latency in the case of Data Centers plays an important role as well. Data Centers contain multiple servers connected by a network, always exchanging information, e.g. delivering a response to a search request [7]: the task is distributed among a group of servers. To minimize the time of delivering of a response, one has to minimize latencies [8] between servers,

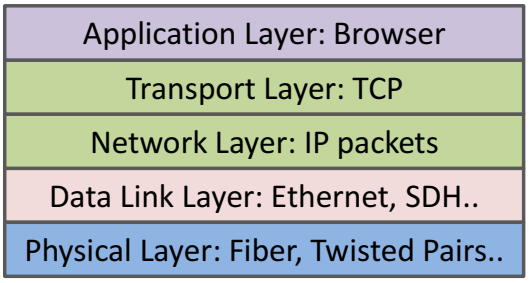


Figure 1.2 – Open Systems Interconnection Model

especially tail (99th percentile) latency [9] when the network load is high. If these strict time sensitive requirements are not met, big revenue losses may happen for service delivery companies, such as Google or Amazon [10]. It is the optical networks that can come to the rescue and deliver the lowest network latencies possible, which are limited only by the speed of light [11].

Apart from bandwidth and latency, structural requirements come into play, such as connectivity and cost. As mentioned above, networks consist of several segments that differ on size and connect all of the next: radio base stations, Data Centers, homes, plants, and businesses located in different geographical regions. When the requirement of connectivity arises, it is an optical network that can answer it: only optical fibers that compose optical networks can provide the impressive attenuation of the signal, such as 0.2 dB/km. This directly means, first, the possibility to connect locations tens of kilometers away without any amplification of the signal and up to thousands of kilometers when amplifiers are used. Second, that means that optical transceivers can send less powerful signals, than in other media, such as electronic or wireless, directly meaning less energy consumption and cost efficiency.

Notwithstanding the role of other contributors, optical networks are one of the most important drivers and enablers of telecommunications. To support the global demands of networks on bandwidth, latency, connectivity, and cost, it is optical technologies and optical networks that can fully answer them.

1.1.2 Layered Structure of Telecommunications

According to the Open Systems Interconnection Model (OSI model) [12], present day telecommunications rely on layered structure of communication protocols, where each protocol is responsible for providing a level-specific link abstraction. Layered structure representation is shown on Fig. 1.2.

Application Layer represents the highest layer of protocols, which are specific for an application that is being used, be it a browser showing a web-page, Base Band Processing Unit (BBU) in C-RAN connected to a Radio Base Station located and processing In-Phase/Quadrature (I/Q) samples of radio signal, or just an File Transfer Protocol (FTP) server. This layer ensures the functioning of an application according to its needs and consists of transporting and managing data as a whole, e.g. a photo of a cat encoded in a specific format. The data is then partitioned and transmitted by protocol specific-data units (PDU), with each layer operating with their specific types of PDU.

Transport Layer provides support for application layer protocols and relies on the network layer. Transport Layer protocols control the transmission of application data

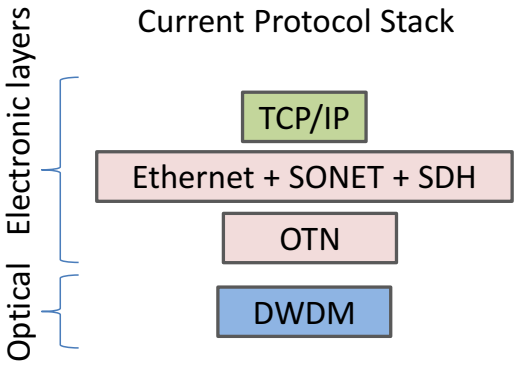


Figure 1.3 – Common Protocols Stack in Optical Networks

partitioned on a transport layer PDUs. Examples of the transport layer protocols are next: Transport Control Protocol (TCP), with TCP Segment as UDP, or User Datagram Protocol (UDP), with UDP datagram as UDP. Each TCP Segment or UDP Datagramm is then encapsulated onto IP packet, a UDP of Network layer.

Network Layer is represented by the IP protocol, which manages individual IP packets whereas Transport Layer manages a set of them. IP packets contain source and destination addresses of nodes, where the application resides. These addresses help to route an IP packet in the network: it may traverse several routers and nodes in the network before will be delivered to the destination. If transport layer provide the end-to-end communications, IP has and hop-by-hop service, with intermediate nodes reading their IP addresses in order to route a packet further.

Data Link Layer defines the way IP packets encapsulated in data link layer PDU, frames, are transported on link-by-link basis from source node towards a destination node that may contain multiple intermediate links. The data link protocol ensures that these links would deliver the data in the form of the frames. Examples of the data link protocol are Ethernet [13] protocol, primarily for Local Access Network (LAN), Synchronous Digital Hierarchy (SDH) [14] or Synchronous Optical NETworking (SONET) [15] protocols for optical MAN and core optical networks.

Physical Layer is the layer that physically transports the data on the communication channel between two or more nodes. The physical layer may represent copper cables, wireless connections, or optical fibers. The frames from Data Link Layer are sent over said media, by modulating some physical property of the media, like amplitude or phase of an electromagnetic wave propagating in the air or fiber, or by modulating the voltage on the copper fiber.

Thesis contribution: This thesis concentrates on the Cross Layer Design of telecommunications, which means that we take into account several layers of OSI model and optimize them into working together beyond what could be done/reached by optimizing layers separately. Such optimization will result in a benefit to some properties of communications, such as latency, energy consumption, or the throughput.

1.1.3 Optical Network Layers Stack Overview

Currently, the communications consist of interdependent layers of protocols depicted in Fig. 1.3 and the majority of them are managed in the Electronic domain,

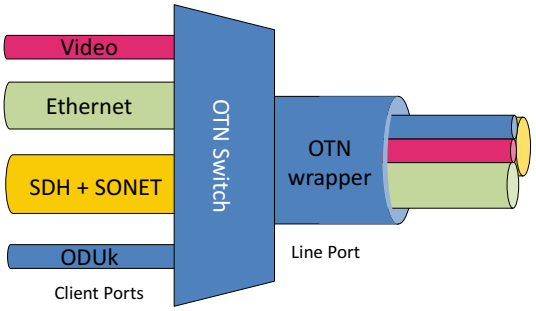


Figure 1.4 – Optical Transport Network (OTN) Protocol Wrapper

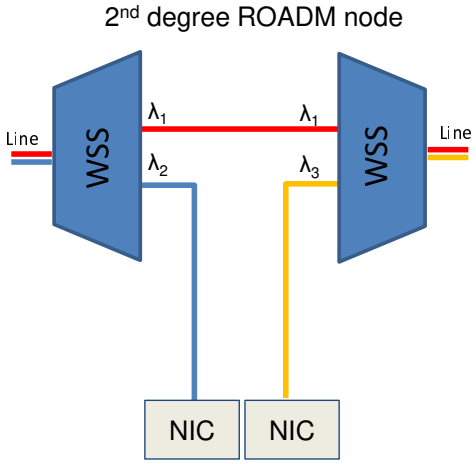


Figure 1.5 – Second Degree ROADM operation principle

whereas only Physical Layer can be managed in the optical domain.

Without the loss of generality one can establish next scheme of the data transmission: TCP regulates the transmission of IP packets, which are put onto Ethernet, SONET or SDH frames, which are then encapsulated onto Optical Payload Unit (OPU) which are then sent in an Optical Transport Network (OTN) [16] (cf. Fig. 1.4). OTN is a set of protocols that relies on the optical media for transmitting the signal but still is managed in the electronic domain. However, the optical media is a specific one, which is represented by Dense Wavelength Division Multiplexing (DWDM) channels.

DWDM [17] primarily consists in transmitting information on up to 80 wavelengths in C-band (1530–1565 nm), where each wavelength behaves as an independent channel. DWDM technologies rely heavily on the use of an optical device, called Reconfigurable Add and Drop Multiplexer (ROADM). ROADMs can separate and multiplex wavelength channels and route them through specific ports thanks to a Wavelength Selective Switches [18]. An example of the ROADM node is represented in Fig. 1.5. There we can see an example with optical channel on λ_1 passing through a node without any Optical-Electronic-Optical (OEO) conversion; and add or drop channels on λ_2 or λ_3 from/on Network Interface Cards (NIC) that convert data from electronic domain onto optical one through Optical Electronic (OE) and Electronic Optical (EO) conversions. This way one can create wavelength specific topologies in the optical network.

The current generation of optical networks has ROADMs, which have optical func-

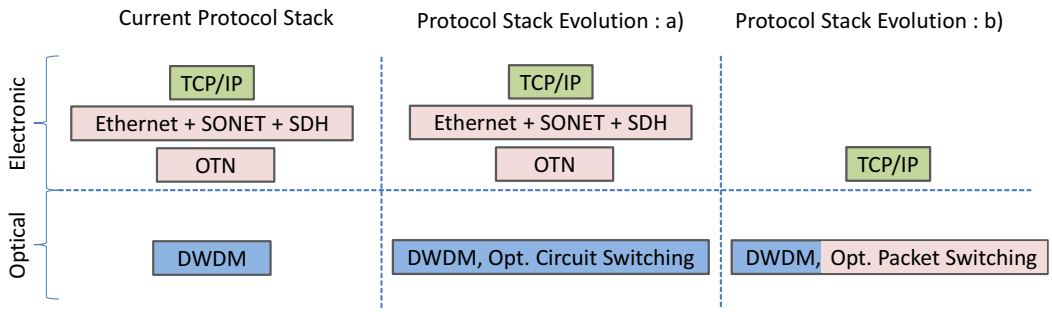


Figure 1.6 – Possible Evolution of Protocol Stack towards All-Optical Networks

tionalities: each wavelength has a specific path in the network and is reconfigurable on command. It is no longer just point-to-point communications as in the previous generation optical networks [19], and now an optical property, a wavelength, contains the information on the wavelength-unique optical path between nodes. However, this could and should evolve further if we would move the upper layers of communications from the electronic domain onto optical.

1.1.4 Possible Evolution of Optical Network towards Hybrid Optical Packet Switching

In the heart of all electronic domain layers lies the technology called Electronic Packet Switching (EPS). EPS defines how the data is routed in the network to achieve its destination. EPS relies on routing the packets of data, where each packet has information about its destination, e.g. IP address in IP packet. When the packet of data enters the router or network switch, the address is read and data is sent on the required output. Depending on the layer, such a packet can have different names and types, for example for the Network layer it is IP-packets, in the case of Ethernet/SONET/SDH/OTN layers it is frames, but the principle is the same and happens always on electronic domain: a switch/router receives a chunk of data, reads the address, decides to which output to send, and sends it.

There is a persistent interest in the scientific community in moving higher layers of communication on the optical level [20]. By going away from electronic domain dominance, one could achieve several major improvements in several characteristics, such as better latency, transparency to advance modulation formats (i.e. possibility to increase traffic volume per fiber), and energy-savings due to disappearance of numerous OEO conversions. But it's easier said than done, which explains the research stage of such kinds of proposals. All of them have something in common: they try to replace the dominant switching mechanism of networks – EPS. Most of the proposals are centered around the introduction of a new type of switching, e.g. Optical Circuit Switching (OCS), Optical Burst Switching (OBS), Optical Packet Switching (OPS) [21] or any other hybrid solution with a mix of this (e.g. EPS) and that (e.g. OCS).

OCS consists of fast switching of optical circuits that are composed of only optical fibers and a light path running on them. The switching happens on the level of the whole circuits. The circuits are created by the configuration of OCS switches, based on Micro-Electro-Mechanical System (MEMS) mirrors [22]. Such systems may well be used in Data Center or High-Performance Computing (HPC) networks [23]. The time of

configuration of a switch is on the order of milliseconds, but the time of configuration of the whole path depends on the number of switches on the light path and may be much higher. The time of tearing down of the circuits as well may take additional time. Due to these constraints of times of setting and tearing down of the circuit, the OCS lose to EPS in throughput if the amount of data to transmit is not big enough, i.e. when the time needed to transmit data is much less than the time needed to establish and tear down the connection. However, OCS let us get away completely from OEO conversions and let to use only one pair of transmitter/receiver, whereas EPS require to use a number of pairs transmitter/receiver equivalent to the number of EPS switches plus one pair for origin/destination server, on the way of the light path.

Optical Burst/Packet Switching (OPS/OBS) [24] may be used as an alternative to EPS. The technology consists in switching of data on the level of packets/frames/bursts on the switches: optical packets or burst are not converted to the electronic domain, and are routed to the switch output directly (switching matrix could be based on Micro-Electro-Mechanical Systems (MEMS) or Arrayed Waveguide Grating (AWG), cf. Ch. 2). OPS and OBS use statistical multiplexing the same way as EPS and does not require the reservation of the whole path as in OCS, leading to better throughput. The key difference between OBS and OPS is that OPS orchestrates the switching of packets that possess only specific and dedicated content, whereas OBS switches bursts of data, which have a set of various data that happen to have the same source and destination address. OPS and OBS supposed to get away from the OEO conversion of EPS and have the same network throughput, but unfortunately, the contention issue rises. OPS and OBS are susceptible to contention issue [25]: when the two packets enter from two different inputs and have to be sent to the same output. In that case, in the absence of optical buffer, both packets are lost, meaning high Packet Loss Ratio (PLR) [26]. High PLR drastically deteriorates the quality of telecommunications.

Hybrid solutions are also proposed to keep OPS/OBS advantages of statistical multiplexing, EPS's advantages of throughput, OCS's advantages of OEO conversion reductions: concurrent use of EPS and OCS [27], or EPS and OPS [28]. We would like to pay attention to another hybrid solution, that consists of addressing the contention problem of the OPS by equipping OPS switch with shared electronic buffer [29] and resolving high PLR problem.

Thesis contribution: We generalize mentioned solution [29] with considering a variable number of buffer inputs/outputs, and call it as Hybrid Optical Packet Switching (HOPS). In the current absence of optical buffers, this solution is viable and helps to maintain all of the advantages of EPS and OCS without losing advantages of OPS. It was shown that the addition of electronic buffer does not entail a significant rise of OEO conversion [30]. We will discuss more precisely this solution in Ch. 2. The performance of HOPS network could be potentially enhanced more if we would apply some tailored to OPS TCP Congestion Control Algorithms (CCAs) [31].

We can conclude that to boost the performance of optical networks, one needs to evolve the current optical protocol layer stack, by moving some of them onto the optical domain as shown in Fig. 1.6. In this thesis, we argue that the best method for this is to opt for HOPS networks and we will study these questions in the next chapters of this thesis.

1.2 Optical Networks for Data Centers

Data Center Networks (DCN) are responsible for the biggest amount of traffic on networks in general. According to the Cisco Global Cloud Index of 2015 [32], the amount of monthly Intra Data Center traffic is more than 8 times bigger than Inter Data Center traffic and is 76 times bigger than the Non-Data Center Traffic. According to the 2018 update [33], such traffic that resides entirely in Data Centers will reach 14.6 ZettaBytes per year by 2021, growing almost by 3 times from 2016.

To support such predominant nature of Intra Data Center traffic and its constant growth, data centers require high performing networks with stringent requirements on various parameters, such as throughput, latency [34], and energy consumption. Over the last decade (plus one year) there were several major innovations in DCNs. M. Al-Fares in 2008 [35] proposed a scalable network topology for support of high performing networks composed of commodity switches, which was adopted by major companies, e.g. Facebook [36]. In 2011 M. Alizadeh proposed to use efficient Data Center TCP (DCTCP) [37] to reduce latencies in DCNs, which was later set as a quasi-standard for DCNs [38]. However, these innovations don't remedy every network performance problem, and all of them are based on use of Electronic Packet Switching (EPS) technology. Current DCNs exhibit sub-optimal Data Center (DC) resources utilization, e.g. CPU, memory, etc., and this time an optical network could be used to address that problem [39].

Optical Network in Data Centers requires advances and progress in network building and performance just as any network, but possess some flexibility, as they represent a controlled environment. A maintainer of a Data Center can adapt the network for specific needs and can build tailored optimized networks consisting of some special equipment, if needed. DC networks necessitates low latency and energy consumption, which makes them as a fertile ground for the application of different types of optically switched networks.

There are plenty of proposals for data center networks that try to diverge from EPS dominance. These solutions based on OPS, OCS, or hybrid solutions as described in the previous section of this chapter. N. Farrington et al. explores extending the use of just of EPS by OCS for high bandwidth traffic [40]. A. Singla et al. propose a network that would rely solely on OCS [41], the benefits of which is that solution is feasible as there is no problem of contention to solve and it let to get rid of OEO conversations completely. N. Benzaoui et al. propose an analog to OPS [42] – Cloud Burst Optical-Slot Switching (CBOSS), CBOSS uses Software Defined Network (SDN) control to intelligently schedule traffic.

There are a lot of solutions proposed, based on what we call Hybrid Optical Packet Switching (HOPS) (cf. Ch. 2), as this technology potentially can have benefits of network performance of EPS with better latency characteristics, energy consumption reduction of OCS, by solving contention problems of OPS. We will speak about them more in the next chapter, cf. Ch. 2.

Thesis contribution: First of all, from the forecast and demand for innovation by the industrial community, secondly, from research activity in the scientific community in the field of optical networks, we can conclude on worthwhile of the investigation of application HOPS to DC. We put this as the objective of this thesis: whether the application of HOPS to DCN is beneficial and more advantageous than other schemes.

1.3 Thesis Structure

This thesis is composed of several chapters, that consists of the incremental study of the application of HOPS in DCN from different perspectives and sides.

This chapter provides a general introduction to Optical Networks, justifies their use for Data Center Networks and motivates the study of the HOPS in DCNs. Chapter 2 provides state of the art in the application of HOPS in DCN, with a review of the most notable OPS and HOPS solutions already proposed to be used in DCNs. Original work was published in [43] and [44]. Chapter 3 describes the general framework for our studies in terms of the context, and assumptions taken. Chapter 4 proposes specially tailored TCP CCAs for HOPS in DCNs, and study the application of HOPS in DCNs in term of throughput. The material in that chapter was predominantly published in [43] and [45]. Chapter 5 studies the application of HOPS from the energy consumption point of view. The results of these studies were originally published in [46]. Chapter 6 discusses latencies achievable in the HOPS DCNs and discusses the use of now-conventional DCTCP, originally designed for EPS, for HOPS. These study findings were originally published in [47]. Chapter 7 considers the potential of the existence of packets of different classes of service in a network and considers class-specific routing rules in a HOPS network. The study was published originally in [48]. Chapter 8 offers general conclusions and discuss perspective work.

1.4 Thesis Contributions and List of Publications

The work presented in this manuscript has led to 2 journal papers cf. Sec. 1.4.1, 5 international conference papers cf. Sec. 1.4.2, 1 national conference paper cf. Sec. 1.4.3 and 1 worskhop poster cf. Sec. 1.4.4.

This thesis considers Optical Packet Switching (OPS) networks vulnerabilities and solutions to overcome them. In particular, we analyze two different level solutions: hybrid switch, i.e. Hybrid Optical Packet Switching (HOPS), representing device level, and Transport Control Protocols with Congestion Control Algorithms (TCP CCAs) representing network level.

This thesis investigates the application of the combination of HOPS with TCP CCAs to Data Center Networks (DCN). Through comparison of known schemes “OPS with TCP CCAs”, “Electronic Packet Switching with TCP CCAs” and proposed new one “HOPS with TCP CCAs”, we are answering what TCP CCAs are adapted to be used for HOPS and why HOPS with TCP CCAs is the best solution to be used in DCN. We study three axes of network performance: throughput [45, 43], energy consumption[46] and latency [47], in order to provide adequate comparison of network types and arrive to conclusion. Furthermore, we study the possibility and benefits of application of class-specific switching rules for HOPS network [48] in DCN.

1.4.1 Journals

— [43] **A. Minakhmetov** C. Ware, and L. Iannone, “Data Center’s Energy Savings for Data Transport via TCP on Hybrid Optoelectronic Switches,” *IEEE Photonics Technology Letters*, vol. 31, no. 8, pp. 631–634, 15 April, 2019.

— [46] **A. Minakhmetov** C. Ware, and L. Iannone, "TCP Congestion Control in Datacenter Optical Packet Networks on Hybrid Switches," *IEEE/OSA Journal of Optical Communications and Networks (JOCN)*, vol. 10, no. 7, pp. B71–B81, Jul 2018.

1.4.2 International Conferences

— **A. Minakhmetov** C. Ware, and L. Iannone, "Data Center's Energy Savings for Data Transport via TCP on Hybrid Optoelectronic Switches," in *IEEE Photonics Conference (IPC)*, TuC3.3.

— [48]**A. Minakhmetov**, C. Ware and L. Iannone, "Hybrid and Optical Packet Switching Supporting Different Service Classes in Data Center Network," in *23rd Conference on Optical Network Design and Modelling (ONDM)*, Athens, Greece: May 2019. To appear.

— [47] **A. Minakhmetov**, A. Nagarajan, L. Iannone and C. Ware. "On the Latencies in a Hybrid Optical Packet Switching Network in Data Center," in *Optical Fiber Communication Conference (OFC)*, no. W2A.21, San Diego, USA: Mar. 2019.

— [44]**A. Minakhmetov**, H. Chouman, L. Iannone, M.Lourdiane and C. Ware, "Network-level strategies for best use of optical functionalities," in *Int. Conf. on Transparent Optical Networks (ICTON)*, no. Tu.B1.3, Bucharest, Romania: IEEE, Jul. 2018, invited paper.

— [45] **A. Minakhmetov**, C. Ware, and L. Iannone, "Optical Networks Throughput Enhancement via TCP Stop-and-Wait on Hybrid Switches," in *Optical Fiber Communication Conference (OFC)*, no. W4I.4, San Diego, USA: Mar. 2018.

1.4.3 National French Conference

— **A. Minakhmetov**, C. Ware, and L. Iannone, "Amélioration du débit des réseaux optiques via TCP Stop-and-Wait sur les commutateurs hybrides," in *ALGOTEL 2018*, May 2018, Roscoff, France.

1.4.4 Students Conferences and workshops

— **A. Minakhmetov**, C. Ware and L. Iannone "TCP Congestion Control in Datacenter Optical Packet Network on Hybrid Switches," in *EPSRC Summer School in Photonics*. University of St Andrews, Scotland: June 2018, poster.

Chapter 2

State of the Art on Hybrid Optical Packet Switching

Packet switching is at the heart of current data networks due to its high flexibility and efficient use of available capacity through statistical multiplexing. However, switching currently must be performed electronically, despite the fact that most of the traffic is transmitted through optical signals. This incurs many Optics-to-Electronics-to-Optics (OEO) conversions, thus a cost in terms of energy and performance bottlenecks of the electronics. Given the traffic's exponential growth, this cost leads to an unsustainable increase of energy consumption and other operational expenses. Optical Packet Switching (OPS) initially has been proposed in 1990s in [49], [50] and gained its maximum interest in mid-2000s [20]. However, with traffic being asynchronous and in the absence of a technology that would make optical buffers in switches a reality, the contention issue rises, leading to poor performance in terms of Packet Loss Ratio (PLR) [26], thus making the OPS concept impractical. To the present moment, several solutions have been proposed to bring the OPS technology to functional level [51], among which we can identify two groups of solutions: network level and device level solutions. This thesis introduces a joint application of two of them: hybrid switches, as a device level solution, and special TCP Congestion Control Algorithms (CCA), as network level solution. We call this joint solutions as Hybrid Optical Packet Switching (HOPS) and view it as a variant of OPS.

The idea of a hybrid switch consists of coupling an all-optical bufferless switch with an electronic buffer [52]: when contention occurs on two (or more) packets, i.e. when a packet requires to use an output port that is busy transmitting another packet, it is switched to a shared electronic buffer through Optical-Electrical (OE) conversion. When the destination output is released, the buffered packet is emitted from the buffer, passing through Electrical-Optical (EO) conversion. However, in the absence of contention (which is the case for most packets), the hybrid switch works as an all-optical switch, without any wasteful OE and EO conversions, offering the possible cut-through operation. Adding a shared buffer with only a few input-output ports lets us considerably decrease PLR compared to an all-optical switch [53], and bring its performance to the one of electronic switches, but now with an important reduction in energy consumption. One would save the OEO conversions for most of the packets: W. Samoud et al. [30] show that for a hybrid switch, that possesses roughly half as many inputs (30) for electronic buffers than switch inputs (64), one can gain more than 50 % in reduction of OEO con-

version in the worst case of 100% load, compared to an all-electronic switch. Thus, such reduction of the OEO conversion could be relayed to significant reduction of the energy per bit, which is always sought by the industry [54].

Additionally to the reduction of the energy per bit, the OPS and its modification, OPS on hybrid switches,i.e. HOPS, could bring the reduction of the latency in Data Centers, as only a portion of the packets passes by the store-and-forward electronic buffer. Knowing that the tails of distribution of latency occurrences plays big role in data centers [54], the OPS and HOPS could be the answer for reduction of latency.

As for use of TCP CCA approach to fight the contention effect, P.J. Argibay-Losada et al. [31] propose to use all-optical switches in OPS networks along with special TCP CCAs, in order to bring the OPS network throughput up to the same levels as in EPS networks with conventional all-electronic switches, negating the effect of the poor PLR of a standalone all-optical switch and ensuring high levels of quality of service of the whole network. In protocol design one could bring forward two main aspects: properly setting the Retransmission Time-Out (RTO) and the size of the congestion window, both of which must be properly set with initialization of every TCP connection and adjusted along its lifetime. RTO is one of the main parameters used to figure out whether we should consider the sent packet as lost and resend it, or keep waiting for the acknowledgement before sending the next packet. When transmission is successful and without losses, RTO is set to a value close to the Round-Trip-Time (RTT), i.e., the time elapsed between the start of sending a packet and reception of the corresponding acknowledgement. The other important aspect of the TCP CCA is the congestion window, i.e. how many packets can be sent before pausing and waiting for acknowledgments. The answer to the question of the evolution of the congestion window depends on the TCP variant and the conditions of the network. According to the authors of [31], in the case of an all-optical OPS network, the TCP Stop-And-Wait (SAW) algorithm, which maintains only one packet in flight, is adapted for Data Centers and LAN networks; whereas TCP modified Additive Increase Multiplicative Decrease (mAIMD) family of algorithms is aimed at making OPS work for larger Metropolitan Area Networks (MAN) networks.

In order to conclude on the importance of hybrid solution for OPS, we are going to review different existing hybrid switch solutions in Sec. 2.2 of this chapter. Use of CCAs for OPS is a part of network levels solutions, which we are going to review in Sec. 2.1 of this chapter.

2.1 Network Strategies for Contention Resolution

The traditional solution against lossy networks has been a Transport Control Protocol (TCP) where the recipient of data packets acknowledges them, so that the sender can realize that some specific packets were lost and retransmit them. An optical switch that loses a packet to contention can extend this technique to all-optical negative acknowledgments (AO-NACKs): somehow sending back a signal that the packet was lost.

This idea was explored in a bufferless variant of LIONS [55]: the switch sends an AO-NACK to a source server so as to demand retransmission of blocked packets. The results have shown that this contention-handling scheme yields a network throughput equal to that of a LIONS switch with distributed electronic buffer. However, it must be noted, that the topology of a reviewed network is a “star” with one switch, while in some

networks, e.g. in data centers, the topology is much more elaborate, and such scheme of AO-NACK might not be viable.

W. Miao et al. [56], in their demonstration of the switch, also considered a variant of AO-NACK scheme with all-optical bufferless switch, in the more sophisticated topology of a data center. Such network exploits several "star" sub-topologies with all-optical switch in a center, with "stars" interconnected through electronic Top-Of-Rack (ToR) switches, and could be referred as one-hop optical network, i.e. no direct connections of optical-to-optical switch. Thus some packets are required to undergo an OEO conversion in order to be routed to destination. In 2018, authors of [56] continued their work and proposed another approach in forming the data center network [57], limiting number of all-optical switches while interconnecting the same number of clusters. However, electronic ToR switches are still being used, requiring OEO conversions for some packets, without network being completely transparent.

The question of extending AO-NACK scheme to a multi-hop optical network was addressed by X. Yu et al. [58]: authors proposed to route back propagating AO-NACK through several switches. However, this requires extra switching capacity (in their case, this means extra wavelength converters). Additionally, by attempting to solve packet contention this way, another problem is created: contention of two AO-NACKs, which is not managed.

P. Argibay-Losada et al.[59, 31] propose another approach: the use of acknowledgments (ACKs), this is the basic principle of conventional TCP, but now applied to optical packet layer ; servers receive ACKs from destination servers, instead of receiving AO-NACKs from switches, and regulate retransmissions through Retransmission Time Out (RTO) timer, upon expiration of which a packet is retransmitted. As well the servers regulate a variable called Congestion WiNDow (CWND), which defines variable number of packets that could be in flight unacknowledged. The rules defining the evolution of RTO and CWND compose a Congestion Control Algorithms (CCA) of TCP. Authors explored two families of TCP CCAs: Stop-And-Wait (SAW) [59] and Additive-Increase-Multiple-Decrease (AIMD) [31], and found out that use of SAW for short-range multi-hop networks and AIMD for long-range multi-hop networks composed of buffer-less OPS switches allows to achieve a network throughput comparable with conventional networks throughput.

We must note that packet contentions are not regulated either only by buffering solutions or by network-level strategies in OPS: there are studies on combination of them. J. Wang et al. [60] explored retransmission strategies in an one-hop OPS network with use of AO-NACK scheme and switches equipped with Fiber Delay Lines (FDLs) as a packet-buffering solution. It was shown that adding a limited number of retransmissions helps to further decrease the packet loss rate.

2.2 Hybrid Optical Packet Switching

In this section we review existing hybrid switch solutions presented in scientific literature, address its general architecture, and explain the assumptions made from the side of hybrid switch for the study presented in this thesis. Mostly, these solutions were applied and discussed in the context of Data Center Networks (DCNs). Summary of solutions presented below is provided in the Table 2.1

Table 2.1 – Hybrid Switch Candidates for Hybrid Optical Packet Switching

Solution	Dimension	Switching Scheme	Contention Resolution
DOS [61]	512x512 (theoretical)	TWC + AWG	Shared Electronic Buffer
HOPR [62]	8x8	TWC + AWG	Shared Electronic Buffer
LIONS [55]	4x4	TWC + AWG	Shared Electronic Buffer Distributed Electronic Buffer Shared + Distributed Buffer No Buffer: N-ACK scheme
HOPR evol. [29]	8x8	B&S + SOA	Shared Electronic Buffer
Hipoλaos [63]	256x256 (theoretical)	TWC + AWG + B&S + SOA	Fiber Delay Lines

2.2.1 Hybrid Switch Proposals

Throughout 2010s there was a persistent interest in finding technological solutions for OPS, exploring various enabling components and architectures, differentiating in contention resolutions schemes and switching dimensions (possibility to switch from N inputs to N outputs).

In 2010 Xiaohui Ye et al. [61] presented a Data Center Optical Switch (DOS), an optical packet switch model, that could be seen as a prototype of the hybrid one: switching was performed through combination of Arrayed Waveguide Gratings (AWGs) switching matrix with Tunable Wavelength Converters (TWC), which essentially performed OEO along with wavelength packet conversion in order to switch it accordingly through AWGs. Similarly to the hybrid switch discussed in this thesis, DOS managed contentions through the shared electronic buffer, storing the contended packets.

In 2012 Ryo Takahashi et al. [62] presented a similar to DOS concept, called Hybrid Optoelectronic Packet Router (HOPR). HOPR, despite its name, was not exactly what we call a hybrid switch, as performed OEO conversions for all the packets by TWC in order to route them.

In 2013 Y. Yin et al. [55] presented low-latency interconnect optical network switch (LIONS), while basing it on DOS and as well designed to be used in DCNs. Leaving the same concept of switching TWC+AWGs authors explored different contention resolutions schemes along with distributed (channel-specific) electronic buffer, mixed electronic buffer (some buffers are shared, some distributed). As well authors of [55] proposed the use of All-Optical Negative Acknowledgement (AO-NACK), where the LIONS doesn't have any buffers, but sends to servers the AO-NACK on the same channel by back propagation, indicating that the packet should be retransmitted. That architecture could be seen as a TCP-related control scheme, however, the reviewed topology is a star, consisting of only one switch interconnecting a lot of servers.

In 2016 T. Segawa et al. [29] proposed an optical packet switch to be used in DCN that is much closer to hybrid switch than HOPR or LIONS/DOS: it performs switching of optical packets through broadcast-and-select (B&S) and then re-amplification by semiconductor optical amplifier (SOA). This switch splits the incoming optical packet into several ways corresponding to output ports, blocks those that didn't match the packet's destination, and then re-amplifies passed packet by SOA. Shared electronic

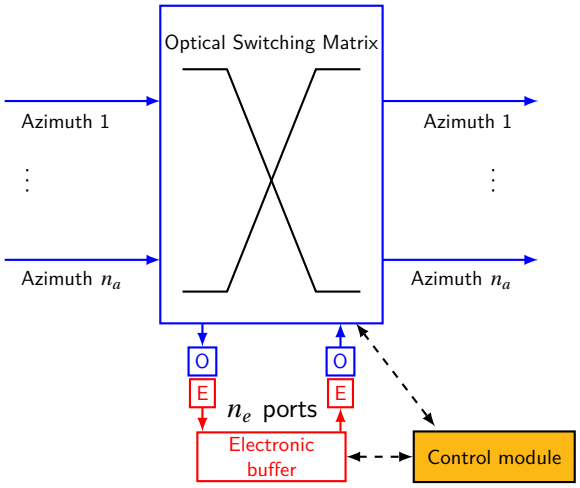


Figure 2.1 – Architecture of a hybrid optical packet switch

buffer solves the contention. The OEO conversion is made only for contended packets, as in discussed in this thesis hybrid switch case, while keeping the non-contended packet in optical domain.

In 2017 X. Yu et al. [58] addressed the issue of back-propagating AO-NACK through several hops, what LIONS were missing due to the star topology, and proposed the concept of solving this issue by introducing TWC on output ports of switch (type LIONS/DOS) for back-propagating AO-NACKs, which may seem effective but still adding to OEO conversions. Additionally, this solution was not numerically evaluated.

In 2018 N. Terzenidis et al. [63] proposed a 256x256 optical packet switch to be used in disaggregated data centers, based on Hipoλaos architecture [39], which used a combination of "B&S+SOA" and AWGs scheme, still undertaking OEO conversion for a packet (by TWCs for use in AWG), however using Fiber Delay Lines (FDL) for contention resolution. FDLs introduce a fixed delay to blocked packets, so as to give a chance to required output port to be liberated and be able to accept blocked packet. FDLs represent all-optical solution, which appears to be more attractive and simple than requiring optical-electrical-optical conversions electronic buffers. However, FDLs may pose a problem with packet latency in asynchronous packet switched networks. Nevertheless FDLs solutions are studied altogether: J. Wang et al. recently [64] provided analytical model for dimensioning optical switch with FDLs and concluded that adding just several FDLs leads to decrease of packet blocking probability.

After this review we can conclude on interesting solution from [29], the only drawback of which is a switching matrix that induces signal losses and requires re-amplification. However this could be overcame by use of recent developments on switching matrices: for example recent T. Chu et al. [65] fast (several ns) 32x32 switching solution, based on Mach-Zehnder Interferometers (MZI) switches arranged in Benes topology. This switching matrix could be potentially applied in 16x16 fully-buffered hybrid switch, meaning equal number (8) of switch and (8) buffer inputs/outputs.

All of the presented solutions above have some common principal blocks, that we are emulating or approximating in our study in order to approach hybrid switch functions. The general structure of the hybrid switch is presented in Fig. 2.1 inspired from [19]. The Switching Matrix switches packets in optical domain without conversion of them to

electronic domain and possess n_a input and output ports (azimuths) to do that. The Electronic Buffer accepts packets that experience contention and then re-emit them using n_e buffer input and output ports. The Control Module manages these two principle blocks. The model of the hybrid switch, that we consider in this study is explained in more detail in the Ch. 3, Sec. 3.2.

2.2.2 Scalability Consideration

When considering switch solutions to be applied in a network, network operators consider the question of scalability: if a solution would support the interconnection of an arbitrary number of servers/nodes, in particular, going from a small number of servers towards hundreds of them. The hybrid switch concept is a scalable solution, providing scalability by two different approaches. Low dimension switches, such as 4x4 or 8x8 can offer scalability in Data Center Networks through use of special network topologies, such as k-ary fat tree clos topologies [35] where k is the number of I/Os of the switch, as in conventional EPS networks with commodity switches. Otherwise, network operators can opt for high dimension switches, as hybrid switches support up to 512x512 any to any switching matrix. Through the combination of these two approaches, the network operators can achieve the required number of nodes/servers interconnected.

The buffer in a hybrid switch requires to use some of the outputs of the switching matrix, thus limiting the network from full use of the switching capacity. As an example, in order to realize 8x8 hybrid switch ($n_a = 8$) with 8 buffer I/Os ($n_e = 8$), one would require 16x16 switching matrix. However, this concern could be overcome by partial use of switching matrix I/Os, e.g. use 12 I/Os as a switch I/O and 4 I/Os for the buffer. It was already shown that just a few buffer I/Os already support acceptable Packet Loss Ratio (PLR) [30], and it will be further shown that 8x8 ($n_a = 8$) switch with just $n_e = 2$ already gives a good performance.

2.3 Conclusion

In this study we decide to focus on and study the combination of two different levels, i.e. cross-layer, of OPS solutions: on device and on network level. As major part of these solutions directed onto use of optical packed switched networks in data center, first of all we aim at the investigation of the potential performance improvements in DCNs and impact of such cross-layer design on the network.

Among all of network level solutions presented in Sec. 2.1 further we will focus an the approach of custom design of TCP CCAs [31], as this solution let use arbitrary topologies of networks, highly customizable and can be easily integrated in the network as multitude TCPs are already used in EPS networks. The only draw-back of such solution is that OPS switches are still subject to contention, and if this effect is remedied, one can potentially achieve better performance of the network.

When considering device level solutions presented in Sec. 2.2, in order to remedy the effect of optical packets contention, we aim at the hybrid switch solution [29], an optical packet switch, equipped by electronic shared buffers. We rely on buffers only in case of contention. Such solution stands out from others because: 1) it is not limited by topology network, 2) it helps to solve contention issue completely, 3) it still has the benefits of optical switching with keeping the majority of packets switched in optical domain.

The study in this thesis is centered around investigating of performance of a Data Center Network (DCN), composed of hybrid switches, where transmission of data is regulated by specially designed Congestion Control Algorithms, making use of the cross-layer solution. Further we will investigate from different sides the performance of such DCN. We start by reviewing the study conditions, context and assumption made in the next Chapter 3.

Chapter 3

Research Framework

In order to study the joint solution of Hybrid Optical Packet Switching (HOPS), we must set up the context, assumptions, and limits of the study. In this study, we are going to simulate the transmission of the data in a data center. A Data Center would consists of a number of servers interconnected through a number of the same-type switches, be it hybrid, all-electronic or all-optical ones. Data transmission would be regulated by TCP Congestion Control Algorithms.

In this chapter, we define the perimeter of the study and provide all the details that help us to create a simulation model that can replicate processes of data transmission in the network and record its performance.

3.1 Optical Packet Switching Model

This study employs the following assumptions: label processing, Control Unit, Switching Matrix are generic and switching time is negligibly small. Optical Packet Switch, or all-optical packet switch, has n_a inputs and n_a outputs, as shown in Fig. 3.1, representing non-wavelength-specific input and output channels, or Azimuths. In-

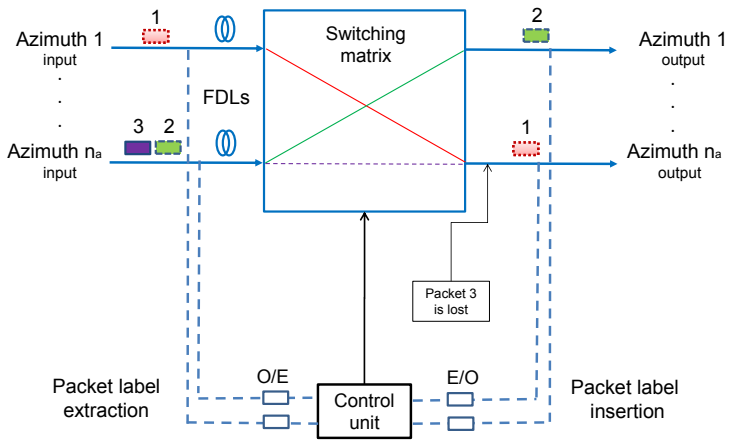


Figure 3.1 – General architecture of all-optical packet switch

put/output pair of the same index represents a bidirectional channel, thus making n_a channels for a switch.

When a packet enters the switch, it carries along a label containing the destination address. Label management is generic so we didn't focus on label extraction, which isn't that easy, but will be required of any OPS/HOPS implementation so can be ignored when comparing them. Nevertheless, we propose and discuss several ways of label management. The labels can be extracted from the packet and processed without converting the packet itself to the electronic domain: the label may be extracted from the communication channel through a splitter (usually 90:10) or a 1x2 MZI switch and then directed to the Control Unit where it undergoes O/E conversion (only the label, and not the whole packet carrying data); or by transmitting them out of band on dedicated wavelengths as in the OPS solution presented by Shacham et al. [66]. This solution allows label extraction via a tap coupler, requiring an OE conversion only for the label as well, and short Fiber Delay Lines at the inputs of the optical switch.

The Control Unit can be represented by Field-Programmable Gate Array (FPGA), which controls Switching Matrix. While the Control unit analyzes the label, the packet is delayed in FDLs so as to give time to the FPGA to adjust the Switching Matrix. Then, it would either route a packet to the desired output, or drop it. If a Control Unit decides to route the packet to desired output, it will generate new label, performing EO conversion, to add it to a packet on switch's output. This mechanism let us to stay out from OEO conversion of the whole packet.

The Switching matrix could be implemented by the technologies described in Ch. 2.2: B&S switch + SOA, TWC+AWGs, or even assembled in Benes Architecture multiple MZIs. Fast switching matrices already exists, achieving fast switching speeds of few ns: switching matrices based on MZIs as in [65] or on SOAs [67]. The optical matrix has a negligible reconfiguration time, on the ns scale [67].

The routing algorithm for the optical packet switch is fairly simple and is the following: a packet enters the switch and checks if required Azimuth output is available. If yes, the packet occupies it. If not, the packet is dropped. On the Fig. 3.1 we show switching process of packets 1,2 and 3. Packets 1 and 2 are switched optically, as their destinations on the switch are available. However, the packet 3 is blocked and lost, as at the moment of arrival the destination on the switch is occupied by the transmission of packet 1. Packet 3 is a victim of the contention process, and contributes to the increase of the Packet Loss Ratio (PLR).

This kind of switch may be referred to as pure OPS and makes the family of OPS switches. The switching matrix is generic and assumes that we can route n_a inputs to any of n_a outputs.

3.2 Hybrid Optical Packet Switching Model

To create a Hybrid Switch we are adding to the essential blocks of all-optical packet switch an Electronic Shared buffer. Same way as other blocks considered previously we assume that Electronic Shared buffer is generic and switching time is negligibly small. Electronic Shared buffer is supposed to be implemented by burst receivers [68]. Hybrid switch has n_a inputs and n_a outputs, as shown in Fig. 3.2, representing non-wavelength-specific input and output channels, or Azimuths. Input/output pair of the same index represents a bidirectional channel, thus making n_a channels for a switch.

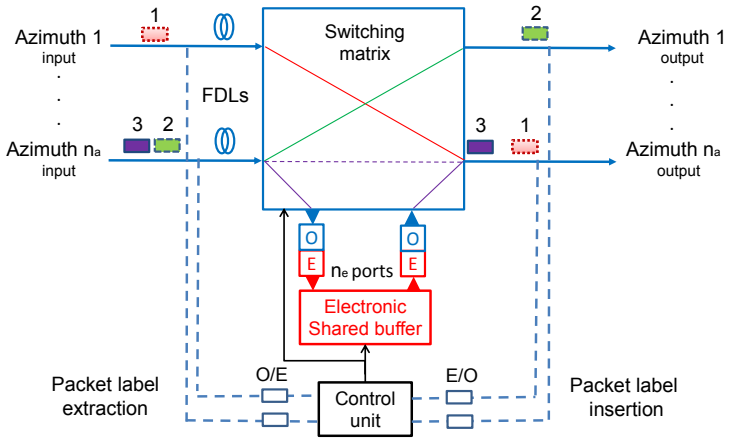


Figure 3.2 – General architecture of hybrid optical packet switch

Another important parameter is n_e: n_e inputs and n_e outputs of a buffer. These are the channels through which the packet is routed/emitted to/from a buffer.

Control Unit is the same as in OPS case, but now also manages Electronic Shared Buffer, and now can route and store packet in a buffer, and then eject the packet from buffer, with new address-label generation.

The routing algorithm for a hybrid switch is the following: a packet enters the switch and checks if required Azimuth output is available. If yes, the packet occupies it. Otherwise, the packet checks if any of the buffer inputs are available. If yes, it occupies one and starts bufferization. If none of the buffer inputs are available, the packet is dropped. If the packet is buffered, it constantly checks if it is first in buffer queue and the required Azimuth output is available; when two of these conditions are met, the packet is emitted from the buffer output and packet occupies the required Azimuth output.

The emission of buffered packets comes along with the rule First-Input-First-Output (FIFO), e.g. if two packets located in the buffer require the same output, then the packet who entered the buffer first will be emitted first. Such emission strategy proved to be the most beneficial according to previous studies by W. Samoud [19]. We must bring the attention to the fact that buffer is shared, meaning that any packet can be routed to any available buffer input among n_e total. Shared buffer helps to decrease the number of transceivers needed for contention resolution in comparison with distributed buffer, where some buffer inputs may be receptive to packets only from specific Azimuths. The beneficial character of the shared nature of the buffer was shown as well in the work by W. Samoud [19].

The case of hybrid optical packet switching is depicted in Fig. 3.2: we show switching of packets 1,2 and 3 the same way as in Fig. 3.1. Packets 1 and 2 are switched optically, as their destinations on the switch are available. However, the packet 3 is switched electronically, as at the moment of arrival the destination on the switch is occupied by the transmission of packet 1, so it is put in the buffer through Optical-Electronical (O/E) conversion. As soon as the packet 1 liberates the output of the switch, packet 3 is sent from the buffer, passing Electronic-Optical (E/O) conversion. Contrary to the process shown in Fig. 3.1, 3 doesn't contribute to PLR, however, this could be still the case, when

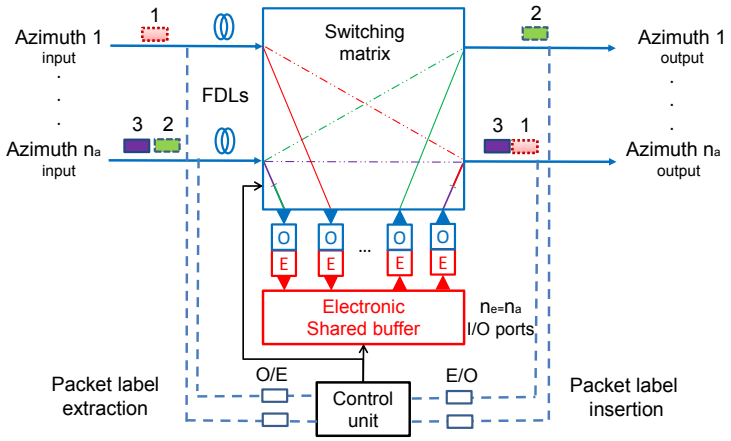


Figure 3.3 – General architecture of all-electronic packet switch

a buffer is occupied with buffering of some other packets, and no other buffer inputs are available (can happen when $n_e < n_a$). Detailed studies of PLR of the hybrid and all-optical switch were undertaken by P. Delesques et al. [21] and by W. Samoud et al. [26], where it was shown that class-specific switching rules can be defined such way to minimize the PLR almost to 0 in high load.

The switching matrix is generic and assumes that we can route $n_a + n_e$ inputs to any of $n_a + n_e$ outputs. In this study, we are going to consider a different number of n_e while fixing n_a according to a topology of DCN, in order to study if and how the performance of the network depends on the number of buffer inputs/outputs.

We note here that presented above Optical Packet Switch is an all-optical switch that is a sub-case of a hybrid switch, but with $n_e = 0$, i.e. in absence of electronic buffers.

3.3 Electronic Packet Switching Model

In order to study the cross-layer design in optical networks form all sides, we must have a reference to a solution that already exists and is working. To reference the application of TCP CCAs on OPS or HOPS networks, we must evaluate the performance of such CCAs also on EPS switches. As well we must consider conventional TCP CCAs developed specifically for EPS networks.

For such a full study, we are going to review the architecture of the electronic packet switch depicted in Fig. 3.3. This switch may be seen as a sub-case of the hybrid switch with specific routing rules: when each packet is routed not directly to the required output, but always first to the electronic buffer and only then to the required destination in terms of switch output. Because in case of EPS the contention is not the issue, we are going to assume that $n_e = n_a$, meaning that the number of buffer inputs/outputs is equal to the number of switch input/output Azimuths.

The switching matrix is generic and assumes that we can route $2 \times n_a$ inputs to any of $2 \times n_a$ outputs.

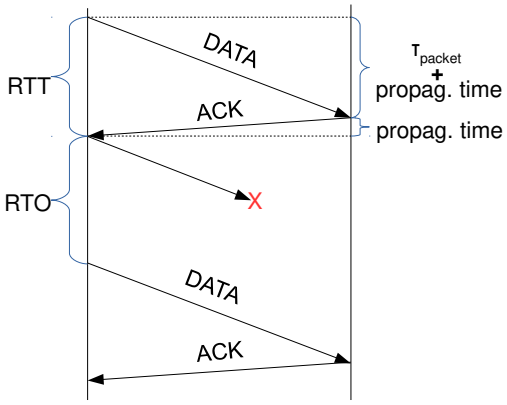


Figure 3.4 – TCP SAW working principle

3.4 TCP SAW – Reference CCA for OPS DCN

While designing protocols in OPS networks one could mostly recognize two cases: when there is at most one packet in flight, and when there might be several. In the former case it is proven to be efficient to use TCP SAW [31] and its mostly adapted scenario is to use it in Data Centers with their relatively short distances. If one decides to use Jumbo Ethernet packet of size 9 kB on 10 Gb/s network interface controllers (NICs) in a Data Center with the longest distance between servers of 600 m, there will be a propagation delay of 2.9 µs [1], which is less than half the 7.2 µs needed to transmit the packet, i.e. packet duration τ. This essentially means that there can be only one packet in flight, and it's more prudent to wait for the acknowledgement before sending the next packet, or retransmits the current packet when the RTO timer expires. Here an RTO T_1 of 1 ms is taken as the initial value, instead of conventional 1 s as suggested by Paxons et al. in [69]. If the corresponding acknowledgement packet is not received within this time, for the retransmission the RTO is now multiplied by a constant factor $\alpha > 1$ so that the RTO is updated as:

$$T_i = \alpha \cdot T_{i-1}, \tag{3.1}$$

up to a maximum value of $T_i = 60$ s. When the acknowledgment is received, the RTO is updated to a weighted average of its current value and the measured RTT γ:

$$T_i = \beta \cdot \gamma + (1-\beta) \cdot T_{i-1}, \tag{3.2}$$

with $\beta \in (0,1)$. In our evaluation we used $\alpha = 1.1$ and $\beta = 0.5$ following the more suitable values suggested in previous works [31]. The choice of the RTO timer's initial value can be justified after consideration of: *i)* the absence of buffering delays in the all-optical network, resulting in a low RTT variance, that is only due to queuing delays; *ii)* the RTT in LANs, that is below of T_1 of 1 ms; *iii)* the RTT in MANs, that is in the order of T_1 of 1 ms. Thus, as long as the destination server is not overloaded with connections, there is no point in waiting for the acknowledgment longer than the true RTT, or a value close to it, in our case the initial RTO. This way the timer helps recover from losses fast enough to maintain a high throughput.

The nature of the TCP SAW is depicted on the Fig. 3.4: the propagation time is so small, compared to the time of the emission of a packet depended on its size, so the time

1. Considering speed of light in fiber with refractive index of $n = 1.45$.

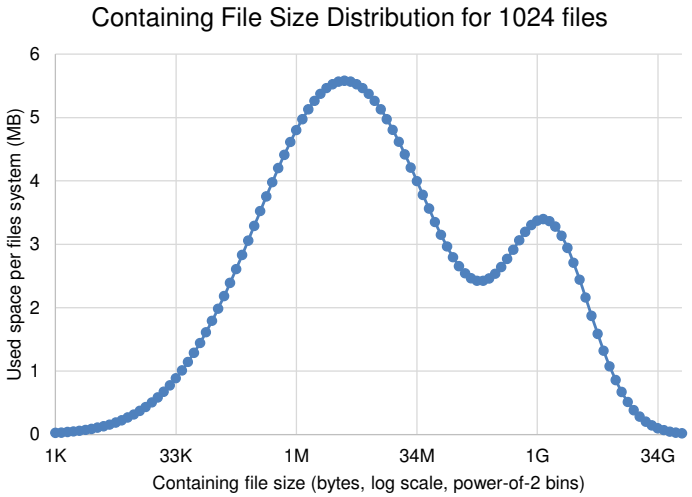


Figure 3.5 – File Size Distribution if to consider 1024 random files.

to receive the acknowledgment containing just a few bytes is almost negligible. This fact helps to detect losses efficiently and re-transmit the packet almost immediately when RTO is close to RTT, which is true when transmission happens without losses the majority of time.

TCP SAW is the reference algorithm for OPS networks that we are going to focus on, as its performance is already investigated and known. In the next Ch. 4 we are going to introduce other TCP CCAs, specifically designed to work in OPS and HOPS networks.

3.5 Simulation of Communications in a Data Center Network

3.5.1 General Scenarios

We simulate the communications of data center servers by means of optical packets, for three scenarios: *i)* when the network is composed of only all-optical switches, *ii)* when it is composed only of hybrid switches and *iii)* when it is composed only of conventional store-and-forward electronic switches. Case *iii)* represents a switch that undertakes whole OE and EO conversions and receives a whole packet, until sending it further. Communications consist of transmitting files between server pairs through TCP connections.

3.5.2 File Transmission Through TCP with Packet Granularity

We simulate transmission of 1024 random files, close to the case of 1000 files in [31] study of TCP SAW, the files' size is random, following a lognormal-like distribution [70], which has two modes around 10 MB and 1 GB, shown in Fig. 3.5. This distribution favors "mice" flows, i.e. small files, over "elephant" flows, i.e. large files.

File transmission is done by data packets of Maximum Transmission Unit (MTU) size [2], i.e., 9 kB. This value defines packet's payload and corresponds to Jumbo Ethernet

2. In our case it defines Maximum Segment Size (MSS) and also Sender MSS (SMSS), cf. Sec. 4.2.

frame's payload. We choose this value so as to be related to previous research on all-optical packet switched networks [31]. A priori the size of the packet will influence the throughput, however we choose the 9 kB to make favorable conditions [59] for all-optical network case while using TCP SAW, and compare its best performance with hybrid and electrical switches.

In our study we also use SYN, FIN, and ACK signaling packets. We choose for them to have the minimal size of the Ethernet frame of 64 bytes [13], and assume that they do not carry any data related to the file content (payload). These packets could carry payload, but in our case they don't, as in the context of the same TCP connection (as it's shown further) the destination server only accepts and do not send any payload back to the source server. The same destination and source servers could change their roles, and start sending a file in opposite direction, but this will be regulated by separate TCP connection. As file transmissions demands (and thus TCP connection demands) arrive independently (cf. Poissonian process further), we can't consider jointly the parallel transmissions of two files in two opposite directions, even between the same pair of servers. We assume that this minimal size would contain all the relevant information about Ethernet, TCP and IP layers, carrying the MAC addresses, TCP flags, Seq and Ack numbers, that are necessary for TCP CCA.

As we still need to attach to the MTU all the information about Ethernet, TCP and IP layers, for simplicity, we just attach to these 9 kB a header containing 64 Bytes discussed previously. Thus we are constructing the 9064 Bytes data packet to be used in our simulations, with a duration τ dependent on the bit-rate. The last data packet of each connection may be smaller than 9 kB in terms of payload, since file size is not an exact multiple of the MTU.

The actual transmission of each data packet is regulated by the TCP CCA, which decides whether to send the next packet or to retransmit a not-acknowledged one. To be realistic, the initial 3-way handshake and 3-way connection termination are also simulated. The network primarily is characterized by the network throughput (in Gb/s) as a function of the arrival rate of new connections, represented by Poissonian process.

3.5.3 Network Switches Simulation

We developed a discrete-event network simulator based on an earlier hybrid switch simulator [53], extended so as to handle whole networks and include TCP emulation.

The simulated network consists of hybrid switches with the following architecture: each has n_a azimuths, representing the number of input as well as output optical ports, and n_e input/output ports to the electronic buffer, as shown in Fig. 3.2. The case of the bufferless all-optical switch corresponds to $n_e = 0$. When a packet is switched to an available azimuth, packet occupies it. If the azimuth is busy, then the packet is redirected to the electronic buffer through an electronic port. The packet will then be re-emitted when the output azimuth it needs is released. The re-emission queuing strategy of the buffer is First-In-First-Out (FIFO) for a given azimuth.

EPS switches have a similar architecture: each switch has n_a azimuths, which also represents the number of input/output ports, but compared to a hybrid switch, the switch buffers all incoming packets, then re-emits them FIFO. In the electronic switch packets are never lost, and all the packets undergo at first OE and then EO conversions.

In all the cases we consider that the size of the buffer, in terms of Bytes it can hold, is not limited. Samoud et al. in [30] indicated that buffer of a hybrid switch is used only by

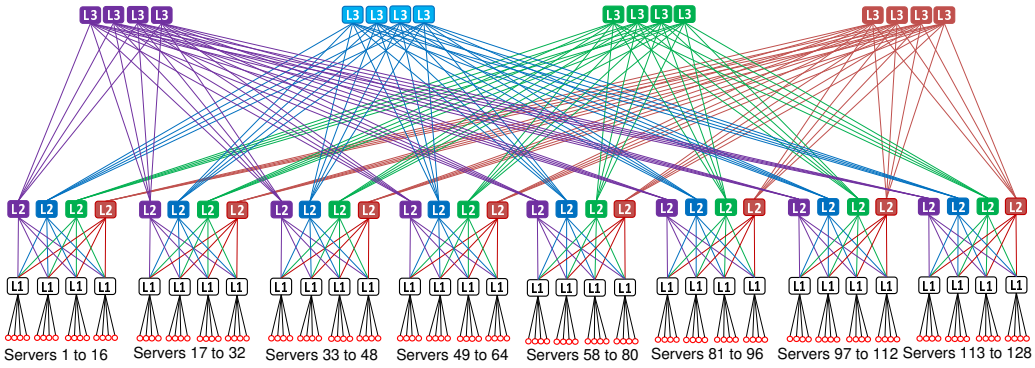


Figure 3.6 – Fat-tree topology network, connecting 128 servers with 3 layers of switches.

a few dozens of packets, implying less than 1 MB for a buffer size.

3.5.4 Note on Fully Buffered Hybrid Switch

In order to compare in a more complete way the case of OPS on hybrid switch versus EPS, we introduce the case, where the number of electronic buffer inputs in a hybrid switch is equal to general number of inputs of the switch. For each optical input port such a switch would have an electronic buffer input, which would be used only in case of packets contention. Cut-through mode of operation would be applicable for general case, and store-and-forward only for contended packets. We call this switch a fully-buffered hybrid switch.

Fully-buffered hybrid switch can be seen as well as an approximation of the high-end cut-through electrical switch. Such hybrid switch offers a cut-through option for non-contended packets, same way as cut-through electrical switch. Contended packets will be buffered in a fully-buffered hybrid switch, almost the same way as in cut-through electrical switch. The difference lies in managing buffered packets: cut-through electronic switch may offer emission to a packet, still undergoing the bufferization process, while hybrid switch would require a completion of bufferization. However, in presence of just two packets in the buffer, requiring the same output (under high load this happens often), even the cut-through electronic switch would be forced to require a completion of packet bufferization, thus negating the difference between buffered packets management. While considering fully-bufferd hybrid switch as an approximation of the high-end cut-through electrical switch, which essentially entails the same performance, we limit ourselves with review of fully-buffered hybrid switch, and do not introduce the cut-through electrical switch. We must note here that fully-buffered hybrid switch entails preferable reduction in OEO conversions, making itself a good candidate for overall energy savings, that cut-through electrical switch misses completely.

3.5.5 Network Topology

The network topology under this study, that interconnects 128 servers [3] by means of 80 identical switches with $n_a = 8$ azimuths, is presented in Fig. 3.6. This is a 8-ary fat-tree

3. As we study transmission of 1024 files, it comes to $1024/128 = 8$ files per server.

topology [35], that represents intra-data center interconnects and is able to interconnect a lot of servers using a switch of just several bidirectional ports (for example, by an optical packet switch of 8 bidirectional ports supporting 10Gbit/s [62]). Switch of level 1 (L1) represents a typical data center top-of-rack (ToR) switch, which interconnects 4 servers of the same rack and 4 switches of level 2 (L2). Switch L2 interconnects 4 neighbor racks and 4 switches of level 3 (L3), which, in its turn, interconnects 8 switches L2, or 8 groups of racks. One can represent the same topology in the terms of pods: there are 8 pods, interconnected by 16 L3 core-switches, in each pod there are 4 L2 aggregate layer switches and 4 L1 access layer switches interconnecting 16 servers. This topology offers for all server pairs the same bisection bandwidth [31] and allow load balancing, as per not-from-the-same-rack servers pair there are several equal paths possible. Such types of Folded-Clos topologies are adopted for Facebook data centers [36], and exactly the same topology as in Fig. 3.6 is considered as one of real-life implemented variants [71].

Each server has network interface cards of 10 Gb/s bit rate. Hybrid switch, presented in Fig. 3.2, is studied with a variable number of n_e, e.g. $n_e \in \{0,2,5,8\}$ of the same bit rate, with $n_e = 0$ representing the all-optical switch case, and $n_e = 8$, that aims to represent a fully-buffered hybrid switch, where $n_e = n_a$, for comparison with electronic switch case.

All links are bidirectional and of the same length l_{link}, e.g. $l_{link} \in \{10,100\}$ m as typical link lengths for Data Centers and LANs. The link plays role of device-to-device connection, i.e. server-to-switch, switch-to-server or switch-to-switch. Link is supposed to represent a non-wavelength-specific channel. Paths between servers are calculated as minimum number of hops, which offers multiple equal paths for packet transmission; that is very beneficial for OPS, allowing lowering the PLR thanks to load-balancing. This means that a packet has an equal probability to use each of the available paths.

In our case we follow a Poissonian process of arrivals of new connection demands between all of the servers: connection demands arrive following the Poisson distribution with a given mean number of file transmission requests per second, which defines the load on a network. The performance of a network with different switches and protocols is studied under progressively increasing load. As in [31] high load could be related to a MapReduce-like model of load distribution in a data-center, supporting a search engine, where network load spikes occur when many servers must rapidly exchange information to form a response, passing through a “shuffling phase”.

To reduce statistical fluctuations, we repeat every simulation with a defined set of parameters a 100 times with different seed for each set of parameters presented above.

3.6 Conclusion

In this chapter we have presented study conditions that we are going to consider while measuring network performance in the Data Center. We have described switching types in the network that we are going to review and a reference congestion control algorithm to be applied, TCP SAW. In the next chapters we are going to present our main scientific contributions, such as: studying the performance of OPS, HOPS and EPS networks with TCP SAW and others, specifically developed during this thesis.

Chapter 4

Congestion Control Algorithms and their Performance in DC Network

In this section we review the performance of the Data Center Network in terms of throughput and introduce new algorithms, adapted for HOPS networks. TCP Stop-And-Wait-Longer (SAWL) is introduced, which is an original algorithm, proposed to enhance performance of TCP SAW [31], developed for all-optical OPS networks, by taking into account existence of buffer in a hybrid switch. TCP Selective ACKnowledgement (SACK) [72], an example of conventional TCP CCAs, used in current EPS networks on TCP level is studied in the context of OPS and HOPS network, with an objective to assess its performance in new type of networks. TCP modified SACK (mSACK), a simplified variant of TCP SACK is proposed and studied here as well. The main application of this CCA is networks with large span.

4.1 TCP Stop-And-Wait-Longer

4.1.1 Algorithm Description

We further develop basic version of TCP SAW, which we described in Sec. 3.4 as the reference for OPS, into a modified one, called TCP SAWL. As we shall demonstrate, TCP SAW works efficiently for all-optical switches, but does not allow to take advantage of the buffers. Indeed, even putting/extracting a packet into/from the buffer adds up to the RTT, becoming longer than the RTO estimated for the previous non-buffered packet, and thus the server will consider such packet as lost. To overcome this limitation, we proposed a modification of the SAW algorithm, so that the RTO is increased by a multiple p of packet duration τ, so as to give a chance to packets having traversed up to p buffers to arrive before the RTO, limiting unnecessary retransmissions. Hence, instead of the T_i shown before, we take as RTO:

$$T_i' = T_i + p \cdot \tau. \tag{4.1}$$

Our simulations consider $p \in \{0, 4\}$, i.e. basic version of TCP SAW with $p = 0$, and one that waits for packets buffered four times with $p = 4$. The essential modification in SAW algorithm is to wait slightly longer, that is why we are referring to it as Stop-And-Wait-Longer – SAWL. We highlight SAW and SAWL differences in Table 4.1.

Table 4.1 – Key differences between TCP SAW and TCP SAWL RTO calculation

Event	TCP SAW	TCP SAWL
If ACK:	$RTO_i = \beta \cdot RTT + (1-\beta) \cdot RTO_{i-1}$	$\mathbf{RTO'_i = RTO_i + p \cdot \tau}$
Else:	$RTO_i = \alpha \cdot RTO_{i-1}$	$RTO'_i = \alpha \cdot RTO'_{i-1}$

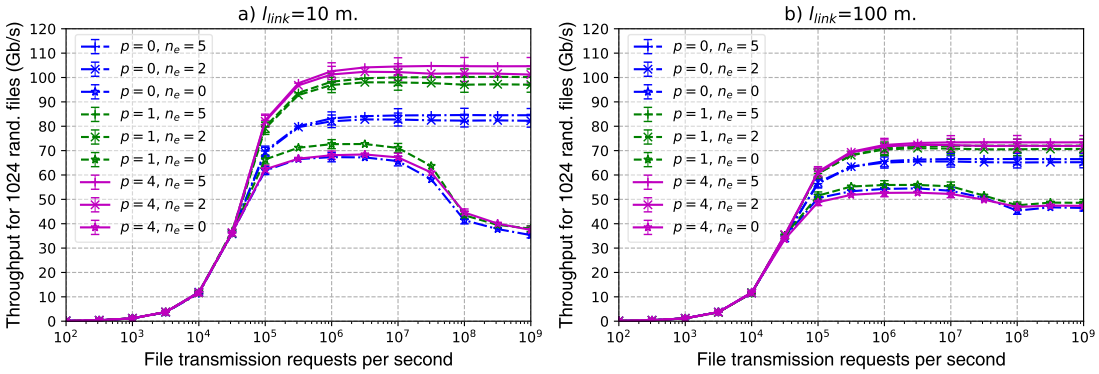


Figure 4.1 – Network throughput dependence on TCP SAWL parameter $p \in \{0,1,4\}$ and number of buffer I/O ports n_e for: a) $l_{link} = 10$ m, b) $l_{link} = 100$ m.

4.1.2 Simulation Conditions

We assess the performance of the TCP SAWL in the context of general conditions described in Sec. 3.5.

We consider the following values of buffer I/Os for switches: $n_e \in \{0,2,5\}$, where $n_e = 0$ correspond to all-optical switch or OPS switch, while other values refers to hybrid switch. In order to understand performance of TCP SAWL we consider the following values for p parameter of TCP SAWL: $p \in \{0,1,4\}$, where $p = 0$ correspond to TCP SAW. Each switch has $n_a = 8$ of switch inputs/outputs pairs. To study how link length influences performance of CCA we consider the following values of link lengths in DCN: $l_{link} \in 10, 100$ m. To reduce statistical fluctuations, we repeated every simulation 100 times with a different seed for each set of n_e, p and l_{link}.

4.1.3 Performance Analysis

The mean throughput for Data Centers and LAN networks with 95% t-Student confidence intervals on every second point is represented Fig. 4.1 to assess performance of TCP SAWL.

The results for SAW in bufferless networks ($p = 0$, $n_e = 0$) differ a little from the results obtained by Argibay-Losada et al. [31], but coincides for high load (more than 10^8 requests/s). This could be explained by the difference in the file size distribution implemented in the simulator, which has a lot of arbitrary parameters, as well as by possible differences in the way of load implementation.

When we consider links of 10 m and 100 m for the case of SAW $p = 0$ and different n_e values, we see that the results do not differ much, except at high load, where the hybrid switch performs better, reaching double the throughput for $l_{link} = 10$ m. The hybrid switch is a robust solution for heavily-loaded networks, so they could support more traffic, in our case after $load = 10^7$ file requests per second.

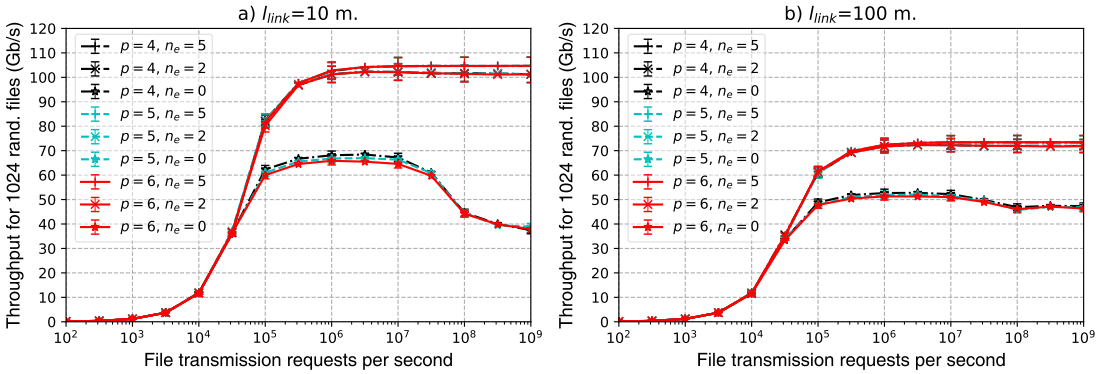


Figure 4.2 – Network throughput dependence on TCP SAWL parameter $p \in \{4,5,6\}$ and number of buffer I/O ports n_e for: a) $l_{link} = 10$ m, b) $l_{link} = 100$ m.

If we consider SAWL ($p > 1$), we see that for data-center networks composed of hybrid switches we have a gain of more than 50% with respect to the bufferless networks using vanilla SAW. The bigger n_e, the better the throughput. At high load (more than 10^7 requests/s) the throughput is increased by a factor of 3 in the case of $l_{link} = 10$ m.

4.1.4 Discussion on the Influence of the l_{link} on Throughput and Latency

Introducing a small modification to the SAW algorithm allows to take advantage of hybrid switches and gain at least 50% in throughput, and even more in some cases. Yet, we still see that increase of link lengths from $l_{link} = 10$ m to $l_{link} = 100$ m leads to decrease of network performance, meaning that propagation time becomes not so negligible compared to packet length τ. Simply put, the link is no longer used efficiently: with $l_{link} = 10$ m the maximum propagation delay for 60 m path is 0.29 µs compared to $\tau = 7.2$ µs leading to minimum Round Trip Time (RTT) of 7.83 µs (tacking into account $\tau_{ack} = 0.05$ µs for 64 B acknowledgement packet); with $l_{link} = 100$ m the same calculations would lead to 13.05 µs of RTT, leaving 5.85 µs for idle time. Such idle time could be used for transmitting other information, but less than a chosen packet size.

Nevertheless, this could be improved – for the networks with the propagation time greater than packet duration, a more suited TCP CCA could be an algorithm from the mAIMD [31] family of algorithms, which is the subject of our following studies. However, on the side note, maintaining only one packet in flight would let us to have the best RTT possible cf. Ch. 6, that would be primarily limited by the distance as in the example proposed above with RTT of 7.83 µs and 13.05 µs for $l_{link} = 10$ m and $l_{link} = 100$ m respectively.

4.1.5 Discussion on the p Parameter in TCP SAWL

During our simulations, we have found that increasing p to more than 4 does not provide better results, meaning that a packet could be buffered up to four times before RTO. We prove that in Fig. 4.2, where we consider the same conditions of simulation as before, but now with $p \in \{4,5,6\}$. We see that performance of network doesn't change in terms of the throughput for hybrid switch. If we consider an all-optical switch case, then we even can see slight performance advantage of case with $p = 4$ over $p = 5$ or $p = 6$.

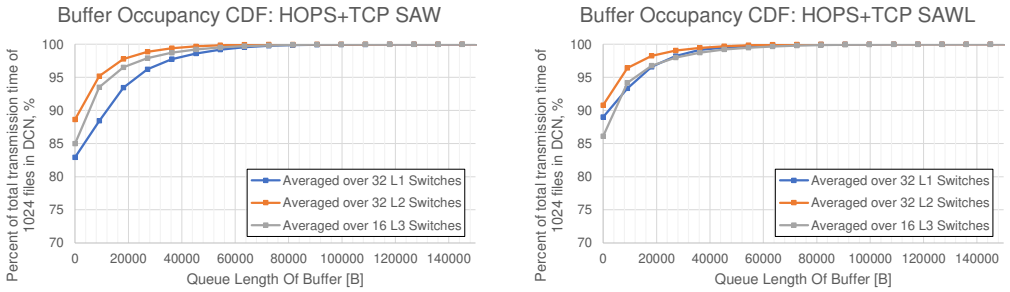


Figure 4.3 – Buffer Occupancy of hybrid switches ($n_e = 8$) of different levels to transmit same set of 1024 files arriving at 10^9 req./s and regulated by: left) by TCP SAW ($kp = 0$), right) by TCP SAWL ($p = 4$)

However $p = 4$ is not necessarily always optimal, the p parameter is linked to the network diameter, i.e. how many switches packet traverses if it goes from one server to another. For the topology under consideration the maximum number is 5, so there is no sense in raising p higher than 5. Setting p to the maximum diameter of network might not be an optimum, as there are still packets that are blocked and not all of packets traverse 5 switches, thus increasing RTO by $5 \times \tau$ for each TCP connection might be counterproductive. In that sense CCA with $p = 5$ is less productive than $p = 4$ with all-optical switch, as RTO is larger and doesn't react quickly to losses. Additionally, for the same reasons we can notice on Fig. 4.1 that all-optical switch with $p = 1, n_e = 0$ outperforms $p = 4, n_e = 0$ (but not in case of hybrid switch: $p = 4, n_e = 5$ performs better than case of $p = 1, n_e = 5$).

One can make value p to vary instead being fixed, but our studies (not shown here) have shown that most of the connections/files have small size ("mice" flow), i.e just several packets, which is not enough to gather enough RTT statistics to make useful conclusions; essentially meaning that life of a TCP connection is too short to "figure out" true p during transmission. This might work if we consider "elephant" flows/files, where life-time of a connection is long enough, but again, proportion of "elephant" flows in a DCN is very low compared to "mice" flows. That is why we propose a simple solution that works for "mice" and "elephant" flows and doesn't require the learning of p.

4.1.6 Buffer Occupancy Evolution in HOPS with SAW to SAWL

In this subsection we review influence of application of TCP SAWL on buffer occupancy of hybrid switch compared to TCP SAW (HOPS, $n_e = 8$). We consider the transmission of the same predefined one batch of 1024 files on load of 10^9 request per second. As a result we measure the amount of time buffer held a certain amount of data, in terms of percent of total transmission time needed to transmit 1024 files in network with $l_{link} = 10$ m. This measurement is presented in 3 categories for different level of switches: L1, L2 and L3 (as in topology presented in Ch. 3). Consequently this metric is averaged among 32 L1 Switches, 32 L2 Switches and 16 L3 Switches.

Measurements in the form of Cumulative Distribution Function (CDF) for a network is provided in Fig. 4.3. For each network we consider next two sets of SAWL parameters: $p = 0$ and $p = 4$. First case is aimed at approximation of TCP SAW (no waiting for a packet to be put in buffer) and second is aimed at representation of SAWL.

First we can witness that buffers of switches no matter what level are empty more that 80 % of time. Second we can see that in case of TCP SAW L1 switches are the most loaded. It becomes no longer the case when considering TCP SAWL: the occupancy of L1 becomes on the level with occupancy of switches L2 and L3, which are almost unchanged. By introducing SAWL we decrease 99th Percentile from 54384 B towards only 36256 B, meaning only 4 full data packets.

Introduction of TCP SAWL let us not only to profit from high throughput in network with hybrid switches, but also to decrease buffer occupancy of L1 switches.

4.2 TCP SACK and TCP mSACK

When distances increase, i.e. in the MAN case, the network is usually able to accommodate several packets in flight. Thus, the use of network bandwidth with TCP SAW becomes much less effective with its single packet in flight, and the throughput decreases drastically. To fight this issue one could prefer to use TCP protocols with variable Congestion WiNDow (CWND), measured in bytes, regulating the number of packets in flight, depending on occurred losses. Authors of [31] propose to use an Additive Increase Multiple Decrease (AIMD) family of algorithms with reduction of the initial RTO towards 1 ms, thus naming it as TCP modified AIMD (mAIMD), as a strategy to control the CWND. All algorithms of mAIMD family share the following general principle: if an acknowledgment of the packet is received before the RTO timer expires, the CWND is increased linearly, and thus more packets can be sent; otherwise, when loss is detected, the CWND is decreased by some predefined factor. Such a strategy helps to achieve better use of bandwidth than TCP SAW and increases the throughput in MAN OPS networks.

In this study we are reviewing the algorithm SACK [72], a candidate for mAIMD, and a modified version of SACK, mSACK, with more aggressive CWND updating. As we review the TCP CCAs together with hybrid switches, TCP SACK is a good candidate for further study as well from point of view of energy consumption.

4.2.1 TCP SACK Algorithm Description

TCP SACK is chosen for our studies as it allows to receive selective acknowledgments (SACKs), i.e. when one packet of the several sent is lost, the server will acknowledge all the packets that successfully arrived to the destination, thus indicating to the sender the missing packets. TCP SACK has three different transmission phases.

Initially it uses the “slow start” phase, where CWND increases exponentially in time when no losses occurred till an $ssthresh$ value (defined on the first packet loss detection and then updated on the course of a transmission). Then, in the absence of losses, algorithm enters a “congestion avoidance” phase: CWND increases linearly.

A “fast recovery” phase is used when loss is detected with reception of 3 duplicate acknowledgments (DUP ACK) of the same packet sent [1], and packets considered to be

1. Acknowledgment is called as duplicate when it acknowledges the same, already fully-acknowledged, data. This amount is defined by all acknowledged data till the first unacknowledged byte. Example: 4 packets are sent, 1st is received and acknowledged successfully. 2nd packet is lost and is not acknowledged. 3rd packet is received and acknowledgement is sent in order to acknowledge reception of data delivered by first packet and third packet. 4th packet is received and acknowledged is sent in order to acknowledge the reception of data consisting of 1st, 3rd and 4th packet. Acknowledges generated by 3rd and 4th represent

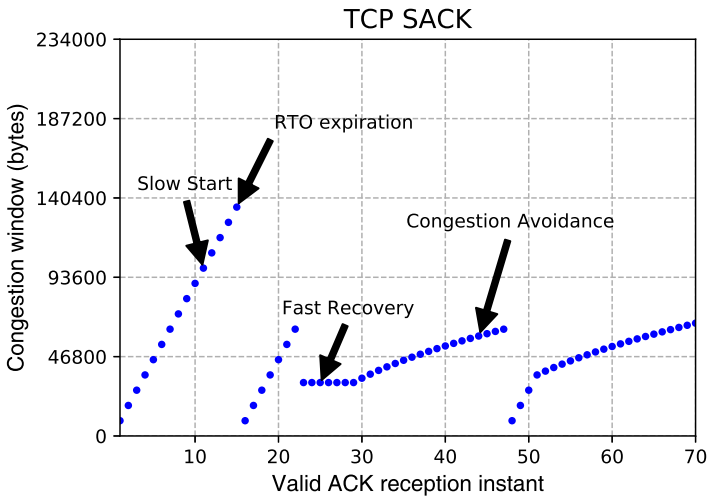


Figure 4.4 – Example of Congestion Window evolution for TCP SACK. Dots represent CWND size that is evaluated each time an ACK is received or RTO times out.

lost are retransmitted before RTO expiration with halving current CWND, and updating $ssthresh$ [73] as follows:

$$ssthresh_i = \max\left(\frac{\text{CWND}_{i-1}}{2},\ 2 \times \text{SMSS}\right), \tag{4.2}$$

$$\text{CWND}_i = ssthresh_i, \tag{4.3}$$

where SMSS stands for Sender Maximum Segment Size and defines the maximum useful payload that a packet can carry. During “fast recovery” TCP helps recover from losses, and doesn’t change CWND. However, if the RTO timer ever expires, CWND is set to 1 packet in flight.

TCP SACK [72] has another phase defined as “rescue retransmission”, which should come during the “fast recovery” phase and allow to retransmit one packet that is not considered as lost. Nevertheless, we opt to omit such phase in our implementation, in order to limit the impact of non-essential retransmissions on network load. As in mAIMD algorithms family, the initial RTO is reduced to 1 ms. We must note here, that in order to make a true comparison to existing electronic switch solutions in Data Centers using different versions of TCP AIMD, for the case with electronic switch we must review not only reduced initial RTO version, but a conventional initial RTO of 1 s.

We present in Fig. 4.4 the example of possible evolution of CWND during various phases for TCP SACK depended from valid ACK reception instant, i.e. CWND change instant, so exponential growth of the CWND during "slow start" in time appears to be linear. We observe the evolution of the CWND under the losses induced by network operation of topology defined in Sec. 3.5 with link length of 100 m consisting of hybrid switches with $n_e = 5$ input/output electrical ports[2]. In our simulations CWND was

duplicate acknowledgment, resulting in 3 acknowledgments, acknowledging the same amount of data from start till the first unacknowledged byte.

2. $n_e = 5$ is chosen to show a case where packet can be lost due to contention in a switch, used in 8-ary fat-tree DC.

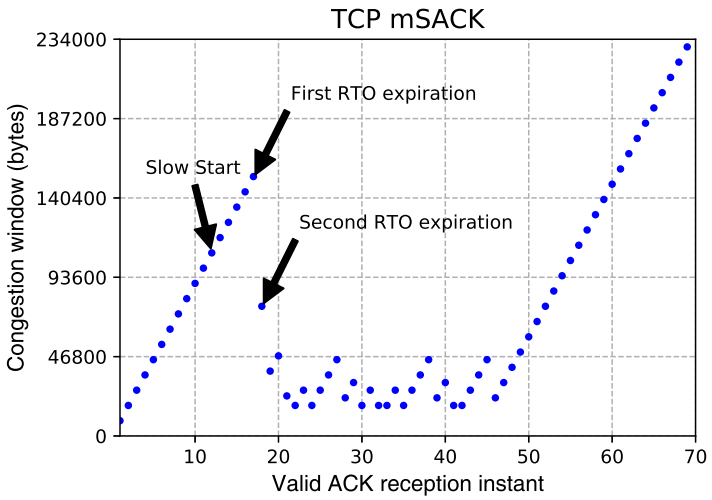


Figure 4.5 – Example of Congestion Window evolution for TCP mSACK. Dots represent CWND size that is evaluated each time an ACK is received or RTO times out.

measured in bytes and is used to determine how many packet of the predefined size could be sent.

As we can witness in Fig. 4.4, TCP SACK retains all the phases described above, it starts with "slow start", with first RTO expiration CWND is decreased to the size of one packet, or SMSS. "Fast recovery" phase starts from reception of 3 DUP ACKs, $ssthresh$ is set as half of previous CWND, as well as new CWND, which is maintained till the end of this phase. As it ends with $CWND = ssthresh$, the algorithm enters in "congestion avoidance" phase, after that we observe again RTO expiration, slow start phase and, eventually, transmission ends on congestion avoidance step.

4.2.2 TCP mSACK Algorithm Description

The modified algorithm TCP SACK, or TCP mSACK, has a more aggressive CWND variant than TCP SACK, without "congestion avoidance" and "fast recovery" phases, which means CWND increases exponentially all the time as long as losses don't occur; also, upon RTO expiration CWND is not reduced to the size of 1 packet in flight, but halved. Receiving 3 DUP ACK doesn't lead to the consideration of packet lost and doesn't decrease CWND. When it comes to CWND management, TCP mSACK exhibits only one principle common to mAIMD algorithms, without advanced phases as in TCP SACK anymore: CWND is Additively Increased when there is no losses and then Multiplicative Decreased upon loss detection. TCP mSACK is suspected to be reviewed in the previous work [31], that initially introduced TCP SAW.

This CCA has the potential in being adapted more to MAN, than to data center networks, as long server-to-server links in MAN has the capacity to fill themselves with a lot of packets. Taking into account aggressive nature of mSACK, it may fill such links faster, than SACK, leading to better throughput. Such hypotheses will be tested in our next studies. However, to make further analysis complete, we include this CCA in this review, and complete its analysis with the EPS case and the fully-buffered hybrid switch.

Considering case of TCP mSACK in Fig. 4.5, under same conditions of network

parameters as in TCP SACK case in Fig. 4.4, we can see, that only RTO expiration changes the CWND, and no "congestion avoidance" or "fast recovery" exists. We bring attention of the reader that direct quantitative comparison of TCP SACK and TCP mSACK cases in Fig. 4.4 and Fig. 4.5 is not applicable, on the contrary to the qualitative comparison that was made earlier, as each case undergoes different number of packet losses at different instants. These packet losses depends on the whole network operation with hundreds connections in it in parallel, and even with all other equal parameters of the network, connections under different CCAs will differently influence other connections, inducing losses at different instances.

As for the case of TCP SAW and TCP SAWL, the congestion window will be always constant and is given by

$$\text{CWND}_i = \text{SMSS}, \tag{4.4}$$

as it maintains only one packet in flight.

4.2.3 Simulation Conditions

We assess the performance of the TCP SACK and TCP mSACK in the context of general conditions described in Sec. 3.5. We as well compare their performance with TCP SAW and SAWL, reviewed previously.

We consider the following values of buffer I/Os for switches: $n_e \in \{0,2,5,8\}$, where $n_e = 0$ correspond to all-optical switch or OPS switch, while other values refers to hybrid switch and $n_e = 8$ corresponds to a fully-buffered switch (cf. Sec. 3.5.4). All-electronic switch, or EPS case, as well reviewed with two parameters of $RTO_{init} \in \{0.001, 1\}s$, where $RTO_{init} = 0.001s$ adapted from OPS and HOPS cases and $RTO_{init} = 1s$ from conventional value in EPS networks. While considering TCP SAW and SAWL we consider the following values for p parameter of TCP SAWL: $p \in \{0,4\}$, where $p = 0$ correspond to TCP SAW and $p = 4$ for TCP SAWL. Each switch has $n_a = 8$ of switch inputs/outputs pairs. To study how link length influences performance of CCA we consider the following values of link lengths in DCN: $l_{link} \in \{10, 100, 1000, 10000\}$ m. Values of $l_{link} \in \{1000, 10000\}$ m, i.e. $l_{link} \in \{1, 10\}$ km are not realistic for DCNs, but we consider them to understand how algorithms perform on increased link lengths.

4.2.4 Evaluation of Results: $l_{link} \in \{10, 100\}$ m

We present here the results of our study and their analysis that will lead us to the conclusion on how the TCP CCAs influence on the throughput on OPS network with all-optical, hybrid and electronic switches. All these reflections let us to conclude on which combination of TCP CCA and switch type is beneficial for which scenario.

The mean throughput for Data Centers and LAN networks with 95% t-Student confidence intervals on every second point is represented in Fig. 4.6 and in Fig. 4.7 for the cases of $l_{link} = 10$ m and $l_{link} = 100$ m respectively. In each figure we represent four cases, one in each subfigure: *a)* TCP SAW, i.e. TCP SAWL with $p = 0$, *b)* TCP SAWL with $p = 4$, as the favorable value we found in Sec. 4.1, *c)* TCP SACK and *d)* TCP mSACK. We shall examine, for each protocol family, the throughput given by hybrid switches and optical switches for all link lengths; then the comparative performance of electronic switches will be studied in the second half of this section.

First of all, we can validate our results for the case with an all-optical network, i.e. $n_e = 0$, under TCP SAW and TCP mSACK (as it is a direct implementation of TCP mAIMD

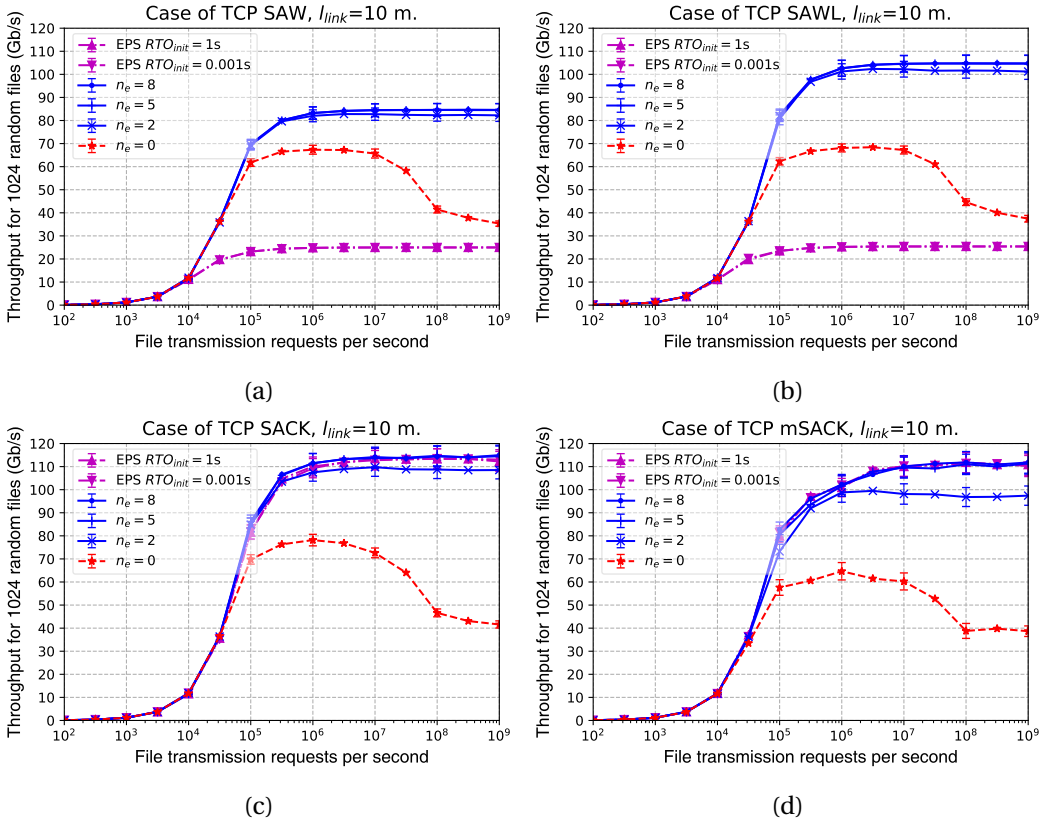


Figure 4.6 – Data Center or LAN network with $l_{link} = 10$ m throughput dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK

algorithm family presented in [31]): the throughput values presented in Fig. 4.6a, 4.7a, 4.6d, 4.7d, have the same order of magnitude and follow a very similar curve for the same numbers of requests per second as those presented in [31]. The slight differences could be explained by differences in the file size distribution implemented in the simulator, as well as by possible differences in the way of a load implementation, i.e. connection demand arrival fashion. Authors of [31] implemented the way of arrival of connections, for the MapReduce application run in a Data Center, when in our case it is a simple Poissonian process. Nevertheless, the results are close enough to validate simulation method.

For the case of the Data Center with hybrid switches paired with TCP SAW in Fig. 4.6a, 4.7a, we can see that hybrid switches already give us better performance than bufferless switches till the overflow load, starting from 10^7 file transmission demands per second, and then keeps its performance, while with bufferless switches the Data Center's throughput drops, so we gain more than 100% for $l_{link} = 10$ m, and more than 30% for $l_{link} = 100$ m at high loads. This could be explained by the fact that the SAW CCA on a server may consider a data packet that was buffered even only once as lost after RTO expiration, and retransmit by sending same data packet. However, the acknowledgment of supposedly lost but only buffered packet is received shortly after retransmission and before its RTO expiration. Then CCA proceeds by sending the next data packet. In

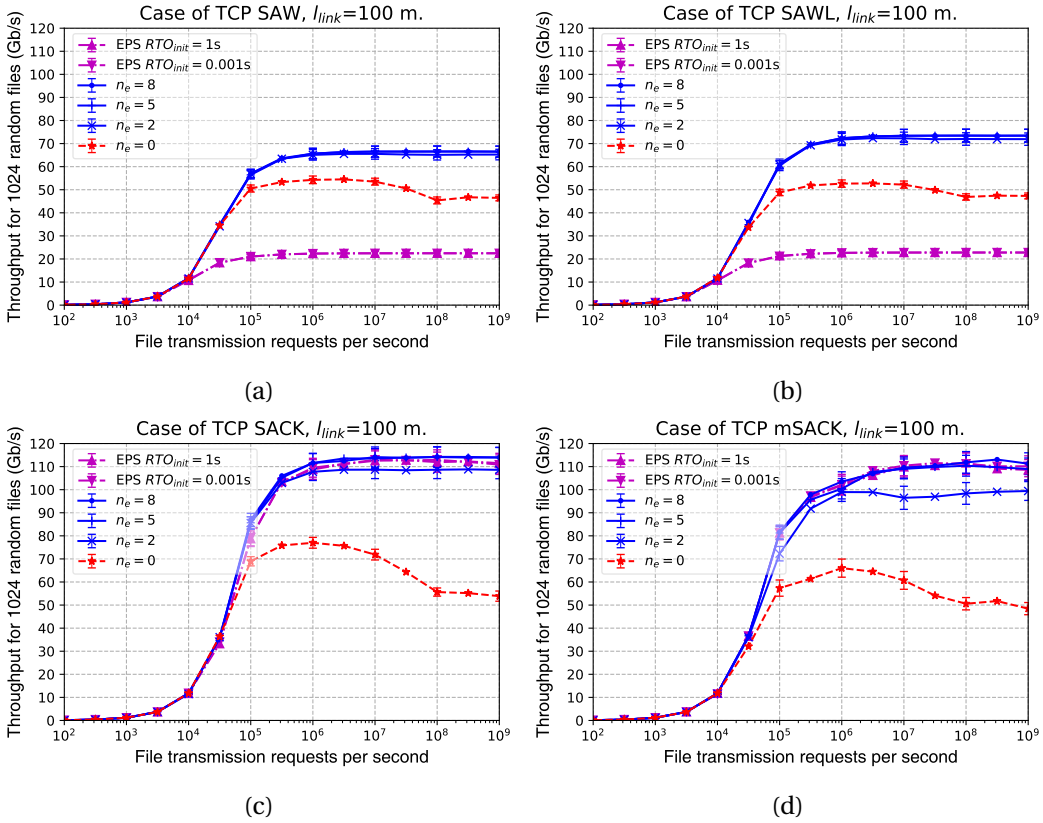


Figure 4.7 – Data Center or LAN network with $l_{link} = 100$ m throughput dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK

all-optical network the next data packet will be sent only upon receiving the acknowledgment of retransmitted packet, as the packet considered to be lost will be lost indeed, thus apparently decreasing the throughput on the overflow load. Such retransmissions in a network with hybrid switches may be considered as wasteful, while in all-optical network they are a necessity.

The higher RTT induced by hybrid switches, but taken into account in TCP SAWL by design to get rid of wasteful retransmissions, lets us increase the throughput for the case of hybrid switch by up to 50% in comparison to TCP SAW for the case $l_{link} = 10$ m (Fig. 4.6b) and by at least 20% for $l_{link} = 100$ m (Fig.4.7b), at high load, i.e. more than 10^5 connection demands per second. The case of TCP SAWL for all-optical network almost doesn't influence the throughput, thus making the gain by hybrid switch paired with SAWL more important: at high load (more than 10^7 requests/s) the throughput is increased by a factor of 3 and 1.5 in the case of $l_{link} = 10$ m and $l_{link} = 100$ m respectively.

In general, for the cases of TCP SAW and TCP SAWL the number of n_e doesn't much influence the throughput. Yet, the bigger n_e, the better the throughput, but already with $n_e = 2$ the average throughput differs only by 5% at worst from the case with $n_e = 8$ under high load.

While reviewing the TCP mAIMD family of TCP CCAs, we can witness in Fig. 4.6 and 4.7 that TCP SACK and TCP mSACK paired with hybrid switches keep its high

performance for the two cases of $l_{link} = 10$ m and $l_{link} = 100$ m and are not sensitive to link length changes.

In comparison to TCP SAWL, as best-performing CCA from the SAW family, TCP SACK and TCP mSACK have same order of magnitude of throughput for $l_{link} = 10$ m and outperform the TCP SAWL by 40% already for $n_e = 2$ in the case of $l_{link} = 100$ m under high load. It is worth mentioning that in the all-optical network case, on the contrary, TCP SAW outperforms TCP mSACK, as shown in [31].

For the cases of TCP SACK and TCP mSACK the choice of n_e plays a more important role than in TCP SAWL. The throughput changes in average by 15% and by 17% in TCP SACK and TCP mSACK respectively under high load.

If one compares the performance of TCP SACK and TCP mSACK for the cases of Data Center with hybrid switches, it is noticeable that in general TCP SACK achieves its maximum at a lower load than TCP mSACK: for example TCP SACK achieves its maximum at 10^6 connection requests per second, when TCP mSACK achieves it only at 10^7 connection requests per second. Also TCP SACK achieves slightly better performance under high load. Yet, TCP mSACK could be regarded as worthwhile candidate for implementation in a Data Center as it could show superior performance on longer links spans, as we will see in the next subsection, and is less complex in terms of algorithmic steps, without "congestion avoidance" and "fast recovery" phases.

In general, no matter what CCA, we can witness that the hybrid switch is a robust solution for heavily loaded networks, which keeps its maximum performance and doesn't saturate as easily as solutions on all-optical switches. TCP SACK seems to be a best candidate for hybrid switches, as it keeps the best possible performance no matter what is the link length in a Data Center network. We considered buffer size as unlimited, but we notice that maximum buffer size depends on CCA, cf. subsec. 4.3.

In order to fully evaluate the performance of hybrid switches, we present the performance of networks with electronic switches, which emulates current existing EPS technology. As was said earlier before, it is important to take into account the conventional initial RTO time of 1 s [69], and not only the value of 1 ms. That said, in Fig. 4.6, 4.7 we can witness this value doesn't influence the throughput in EPS. This could be explained by low losses in EPS, incomparable either to all-optical network or to the cases of network with hybrid switch with $n_e < n_a$, as we simulate buffers of limitless volume. These low losses lead to the fact that retransmissions almost never occur, and each packet is sent only once.

When we review the performance of a network with electronic switches, under TCP SAW and TCP SAWL, one can witness that even all-optical network has much superior performance. This is explained by the fact that all-optical switches operate in a cut-through mode of operation, while electronic ones operate in a store-and-forward way. Thus we add to RTT several times the duration τ, the number of which depends on how many switches packet will see along the way from source server to the destination server. Higher RTT will lead to poor performance when there is only one packet in flight.

However, this is not true for TCP mSACK, where the throughput of the network consisting of electronic switches achieves the same throughput as network with fully-buffered hybrid switches, i.e. when $n_e = n_a$. As for TCP SACK, the throughput of a network with electronic switch is only by $1\% \sim 2\%$ less than throughput of the network with hybrid switches. This could be explained as well by store-and-forward mode of operation in all-electronic case, while in hybrid switch there are some packets that are switched directly to the output, i.e. experience cut-through mode, however some still

stored in buffer, i.e. experience store-and-forward mode of operation. We note, that even with small number of buffer input ports, e.g. $n_e = 2$, hybrid switch let us achieve throughput, which is close to a case with electronic switch case. That fact let us conclude on possible use of hybrid switches. Additionally, even with the fully-buffered hybrid switch, that entails a reduction of OE and EO conversions: Samoud et al. in [30] already indicated that a hybrid switch OEO-converts only a small proportion of packets, which ought to give a significant advantage in energy consumption, though we haven't yet determined how these results scale over the whole network.

Apart from the results that we presented here, we tested as well the case with conventional initial RTO time of 1 s [69] for hybrid switches under different TCP CCAs. The results of such tests had shown significant drop of the throughput under the high load for all-optical and hybrid switches (except for fully-buffered hybrid switch). Such behavior is the result of influence of all-optical part of hybrid switch and, as it was shown in [31] for all-optical switches, initial RTO of 1 s is far from optimal. This fact let us to conclude that reduction of initial RTO towards 1 ms is a crucial parameter of TCP CCA when it comes to application to network consisting of hybrid or all-optical switches.

In general it could be concluded that TCP CCAs enable the use of hybrid switches in data center networks and their specific design is a necessity while implementing new OPS solutions: it is crucial to adjust initial RTO to 1 ms; while using SAW family of CCA, it is crucial to adjust the RTO calculus by several packet durations, so as the system wouldn't waste its resources while performing unnecessary retransmissions.

4.2.5 Evaluation of Results: $l_{link} \in \{1, 10\}$ km

In order to understand how TCP CCAs behave when the length of links increases, we undertook simulations for link lengths of $l_{link} = 1$ km and $l_{link} = 10$ km. Those link lengths are not very realistic for the presented topology, but allow us to get the idea on how the link length influence throughput. We present our findings on Fig. 4.8 and Fig. 4.9.

When we use TCP SAW or TCP SAWL with $l_{link} = 1$ km, we can witness a low throughput for both cases of hybrid and bufferless switches, and there is no difference between the mentioned cases. If we go further with $l_{link} = 10$ km, be it hybrid switch or all-optical or all-electronic, no matter what switch or what type of SAW we use all performance is the same for a fixed link length. This is explained by the fact that for each connection there is only one packet in flight, limiting the throughput, while individual switches' load never rises high enough to cause significant contention of packets. We confirm our previous conclusion as well that the longer the link length, the lower the throughput.

TCP SACK and TCP mSACK exhibits cases when there are much more packets in flight, so that long links can be sufficiently "filled" up with packets: the longer the link physically, the more packets it could contain at the same instant, e.g. a link of 6 km (6 hops of 1 km) can contain more that 4 packets and a link of 60 km over 40.

mAIMD family of protocols keep their performance on the more or less same level till $l_{link} = 1$ km, as we can see on Fig. 4.6, Fig. 4.7 and Fig. 4.8. However, when considering a case of $l_{link} = 10$ km on Fig. 4.9, we see that TCP mSACK performs much better, than TCP SACK: for all cases of switches (all-optical, hybrid or all-electronic) it outperforms it by a factor more than 10 on the average load of $10^5..10^7$ file requests per second. This is explained by much more "aggressive" behavior of updating CWND by TCP mSACK: it

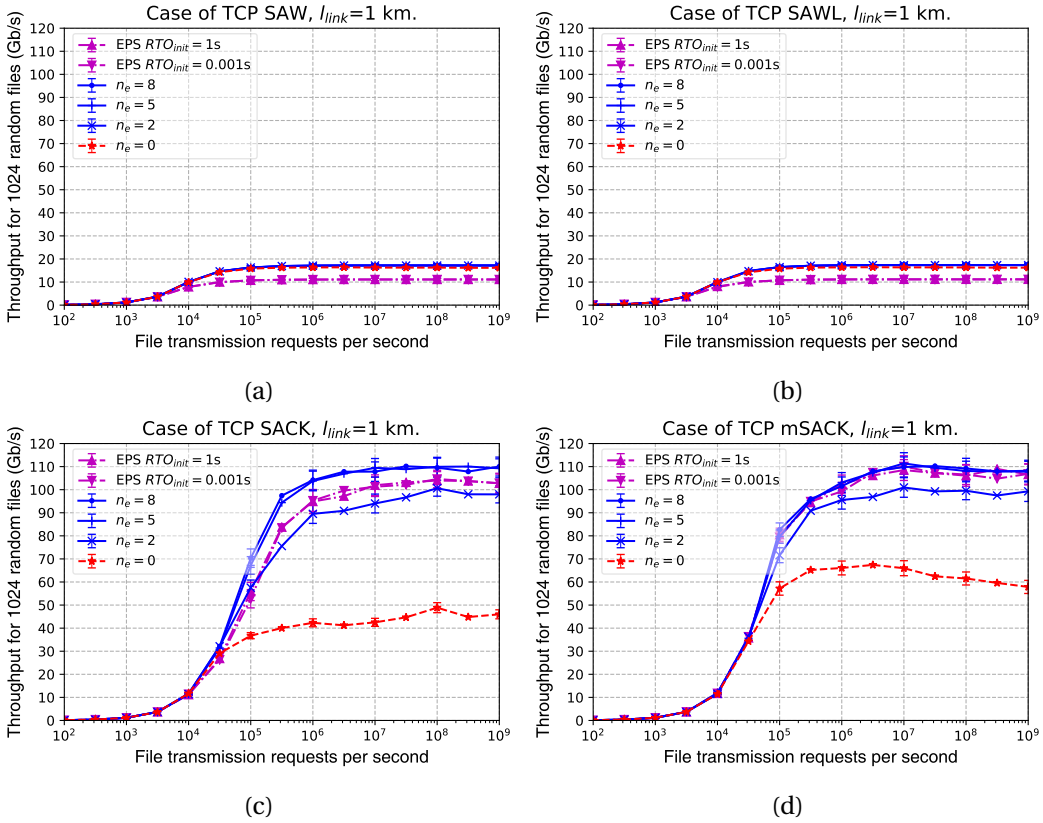


Figure 4.8 – Data Center or LAN network with $l_{link} = 1$ km throughput dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK

doesn't have Congestion Avoidance and Fast Recovery phases (cf. Fig. 4.4 and Fig. 4.5), which limits the growth of CWND in case of packet losses.

With the fact that TCP mSACK outperforms TCP SACK on extra long links we still think that SACK is more preferable to use it on short links, that are realistic for Data Center Networks, and is exhibits advantageous over mSACK behavior.

4.3 Buffer Size of a Hybrid Switch and Its Influence on Latency

In this section we present maximum buffer size that may happen in a DCN considered in this study. We consider the same conditions and context of network simulation as in previous Sec. 4.2. We bring attention of the reader that we focus our attention here on max buffer size occurred, to help to dimension the needed buffer size, and not on the buffer occupancy, as in Sec. 4.1.6 for TCP SAWL or further in Sec. 6.2.5 for TCP SACK, which is statistical measurement.

If we measure maximum buffer size occurred during a simulation, and then take average among all the random seeds, in the worst case for hybrid switches we obtain the next results: SAW – 0.16 MB, SAWL – 0.16 MB, SACK – 1.15 MB and mSACK 5.5 MB. We notice that in general such values decrease when l_{link} increases, that is why we present on Fig. 4.10 measurements only for the case of $l_{link} = 10$ m.

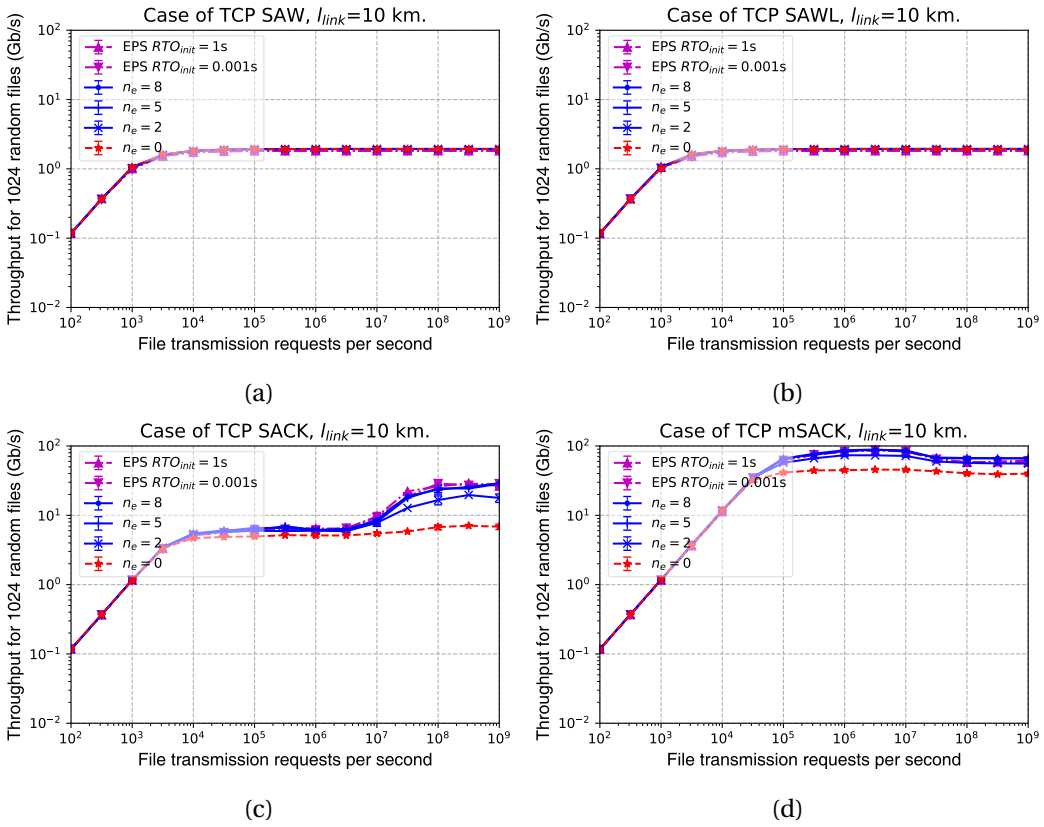


Figure 4.9 – Data Center or LAN network with $l_{link} = 10$ km throughput dependence *in log-scale* on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK

First we can validate that the bigger the load, the bigger the maximum buffer size occured during simulation, which is expected, as the bigger the load, the higher chance of contention of packets resolved by buffer. Biggest buffer use occurs on a network with all-electronic switches, less on network with hybrid switches, and is not used at all on all-optical network.

Second we see that family of SAWL protocols (TCP SAW on Fig. 4.10a and TCP SAWL on Fig. 4.10b) requires much less space than the family of mAIMD protocols (TCP SACK on Fig 4.10c and TCP mSACK on Fig. 4.10d), no matter the type of the switch with the exception of all-optical type of switch that doesn't have buffer. If we consider hybrid switch case, then the difference of the worst case on TCP SAW (SAWL requires less space in general) and worst case of TCP SACK is about a factor of 10. Such difference turns into a factor of more than 30 with the case of TCP mSACK.

Third, we notice that introduction of SAWL has positive effect on buffer size of hybrid switch also, and not only on the throughput as we saw earlier, with comparison with TCP SAW. This difference is better seen for the case of $n_e = 2$, the bigger the number of buffer inputs/outputs, the less the difference is, but still beneficial.

Fourth, we witness that use of TCP mSACK leads to extensive use of buffers, compared to TCP SACK. Fully-buffered hybrid switch with TCP mSACK occupies more than 5 times bigger buffer size on high load than TCP SACK. Such fact tells us that TCP mSACK loses

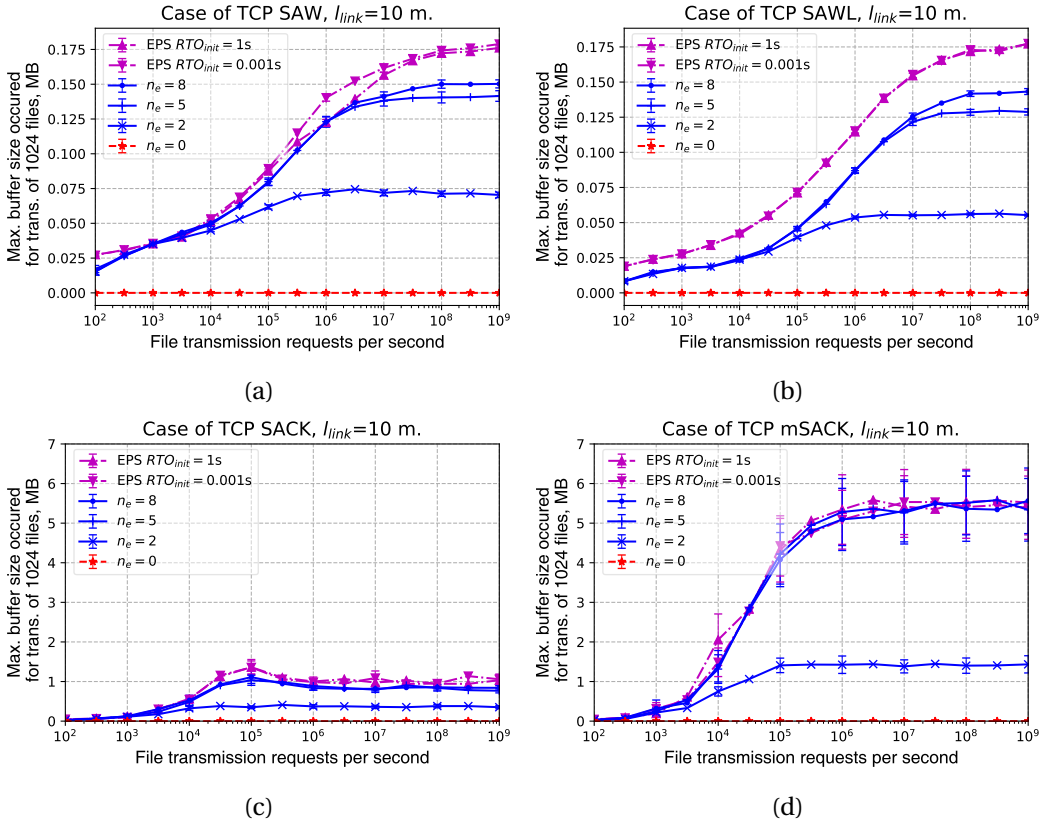


Figure 4.10 – Maximum Buffer Size occurred during transmission of batch of 1024 Random Files, averaged. Dependence on either number of buffer I/O ports n_e or electronic switch with different initial RTO timer for TCP: a) SAW , b) SAWL , c) SACK , d) mSACK

in network latency, as a lot of packets will pass through the buffer, adding up time of adding/extracting and spent in buffer.

Fifth, we can see that the less number of buffer inputs, the less space is used by the buffer, e.g. for TCP SAWL (Fig. 4.10b) for $n_e = 2$ has almost 3 times smaller of buffer size than fully-buffered hybrid switch ($n_e = 8$), at the same time almost the same throughput as seen in previous subsection on Fig. 4.6b.

We can conclude that TCP SAWL is the most favorable CCA among CCAs considered from the point of view of buffer size use, and still allow us to profit of high throughput on network, that uses hybrid switch with only $n_e = 2$ buffer inputs/outputs.

We will be considering question of Latency in Data Center Network further, in Ch. 6, but here we also briefly present resulted latency in Data Center Network in order to corroborate results on Buffer Size under different TCP CCAs. We present 99th Percentile of RTT obtained in Data Center Network with $l_l ink = 10$ m and use of fully buffered switch ($n_e = 8$) in Fig. 4.11.

We see that resulted 99th Percentile on different TCP CCAs repeats conclusions made before based on buffer size. TCP mSACK has the biggest maximum buffer size and results in biggest 99th Percentile, differing by factor of 10 from TCP SACK. At the same time TCP SAWL again shows itself as most favorable among other CCAs, and, specifically, performs better than TCP SAW, as manages to decrease load on buffers. TCP SAW in

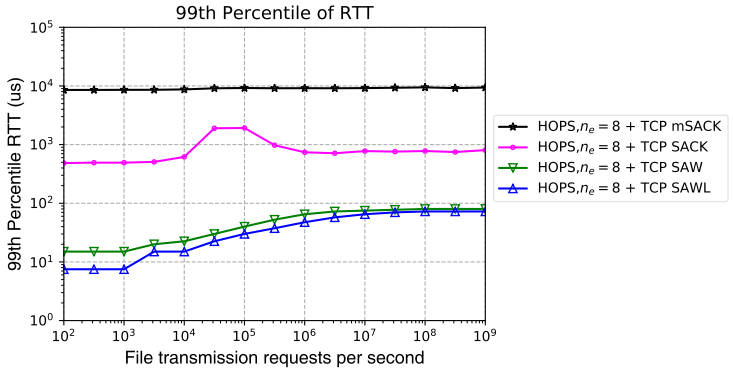


Figure 4.11 – 99th Percentile of RTT for the $l_l ink = 10$ m, of Hybrid OPS ($n_e = 8$) network, depended on TCP CCA

its turn performs better than TCP SACK, as expected from the maximum buffer size measurements.

4.4 Conclusion

The Data Center networks could benefit from hybrid switches that have lower energy consumption than electric ones, and a higher throughput and robustness than all-optical ones, using just a few electronic ports and introducing the specially designed TCP protocols. This could be applied to any type of networks and load by changing the number of the electric ports and carefully adjusting the TCP algorithm.

In this study we completed our previous analysis by reviewing the case of networks with electronic switches under the same conditions (including TCP CCAs) as networks with hybrid or all-optical switches. We successfully showed that, under TCP SACK, a Data Center network with fully-buffered hybrid switches gives even better performance than EPS networks; and a network with hybrid switches even with fewer buffer input ports shows already close to EPS performance.

We demonstrate that direct realization of mAIMD algorithm, TCP mSACK, is not beneficial the same way for short links as TCP SACK, as additionally it exhibits excessive use of buffer of electronic and hybrid switches. Nevertheless exactly this type of CCA outperforms others on extra-long links in the network.

We show that TCP SAWL, a CCA introduced in this thesis, paired with use of hybrid switch for the case of short links, exhibits very close to EPS performance in throughput, use the least space in buffer, and all of that with only $n_e = 2$ buffer inputs/outputs.

Thus, hybrid switches enable us to have a much higher network throughput than bufferless ones in a Data Center network. Pairing TCP SACK or TCP mSACK with hybrid switches becomes a robust and beneficial solution against implementation problems of OPS. Hence, it can be inferred that the necessary condition for the interest in OPS to regain momentum [20] is found.

Studies presented in next chapter are dedicated to evaluating the gain in actual energy consumption of a Data Center under hybrid switches in OPS compared to EPS. All CCAs reviewed here can potentially have different energy consumption, specifically due to the different number of retransmissions under different protocols.

Chapter 5

Energy Consumption in DC Network on Switching

The Optical Packet Switching (OPS) seemed to be a natural step of evolution from Electronic Packet Switching (EPS) in data networks not only because of high reconfigurability and efficient capacity use, but as well due to possibility to curb endless energy consumption growth induced by EPS in attempt to answer growing bandwidth requirements. However, as we saw earlier, in the absence of a technology that would make optical buffers in switches a reality, the contention issue rises, leading to poor performance in terms of Packet Loss Ratio (PLR), thus making OPS impractical and depriving from opportunity to profit from energy consumption benefits.

In previous chapters we have introduced Hybrid OPS (HOPS),which is a combination of two approaches to bring OPS technology to a functional level: hybrid switches and special TCP Congestion Control Algorithms (CCA). We concluded that the throughput of data center (DC) networks can benefit from this combination of TCP CCAs with hybrid switches even with few I/O buffer ports, but our studies didn't analyze possible energy savings, compared to EPS, by having fewer OE/EO (OEO) conversions. In this chapter we aim to address this matter.

It has been shown [74] that transport and switching can represent up to 60% of the total energy consumption in a private cloud storage service, and introducing optics in an EPS network can save about two thirds of the power [75]. An evolution from EPS towards OPS could lead to further improvements in energy consumption through limitation of OE/EO conversions on switches' I/O ports. Taking into account that a transceiver (potentially a switch's I/O port) of 10 Gb/s can spend over 80% of its power on light emission related processes [76], one sees possible energy savings with hybrid switches, that would use these transceivers only for their buffers and not on their main I/O ports. The same conclusion from [76] shows that among OE and EO conversions it is the latter that contributes more to energetic budget, and this is why we choose to base our study measurements on it.

In our work, we consider the following energy-consuming data transport operations to be EO conversions: initial emissions from the servers, reemissions by hybrid switches' buffers and EO conversions by I/O ports of electronic switches.

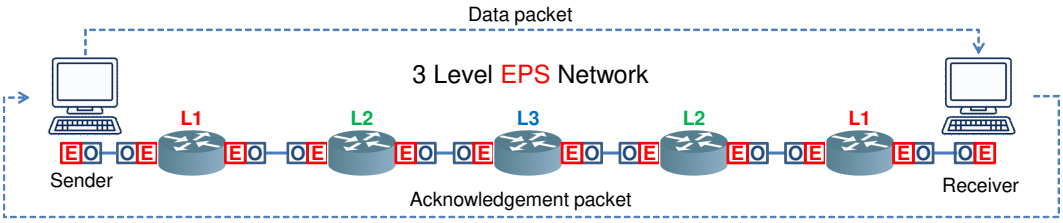


Figure 5.1 – "Bit transport energy factor" for 1 packet in EPS network: $\frac{9064\times6+64\times6}{9000} = \mathbf{6.085}$

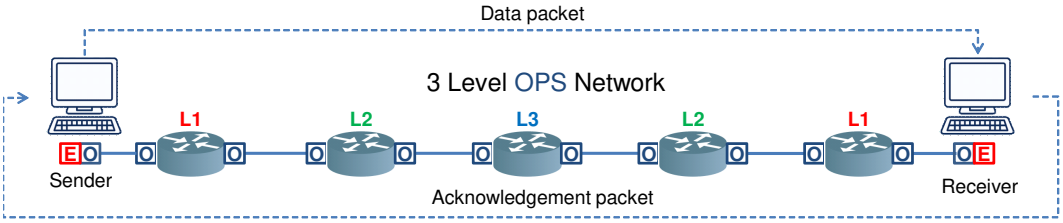


Figure 5.2 – "Bit transport energy factor" for 1 packet in OPS network: $\frac{9064\times1+64\times1}{9000} = \mathbf{1.014}$

5.1 On the Metric for Energy Consumption For Data Transport

The choice of the metric to quantify energy gain due to reducing the number of EO conversions must be made carefully. Measurement of the EO conversions themselves might be not sufficient, as conversion of packets of different sizes will consume a different amount of energy. Measurement of bits that undergo EO conversion may be also not relevant, as it will depend on the amount of data sent on the network. Measurement of power spent on EO conversions of information bits would require choosing a specific emitter, but at the same time we want to stay as general as possible. Therefore, we choose to measure "transmission energy cost" in units of "bit transport energy factor", i.e. how many bits should be physically emitted to ensure the delivery of one bit to the destination. One can think of this value as the number of bits passed through EO conversions within the whole network normalized by the number of total bits that are to be sent into the network over a period of time. For one acknowledged packet it could be calculated according to next formula:

$$\text{Bit transport energy factor} = \frac{Data_{pkt} \times EO_{data} + Ack_{pkt} \times EO_{ack}}{\text{Payload}} \tag{5.1}$$

Where $Data_{pkt}$ is the size of the Data packet consisting in header (always 64 B) and Payload (up to 9000 B), Ack_{pkt} is the size of the Acknowledgment packet consisting just of header. EO_{data} is number of times data packet undergoes EO conversion. EO_{ack} is the number of times acknowledgement packet undergoes EO conversion.

Examples of the "Bit transport energy factor" for one packet transmitted and acknowledged in EPS and OPS networks are presented in Fig. 5.1 and in Fig. 5.2 respectively.

We opt for this measure which represents the energy consumption for data transport in the whole network, which can be converted to J/b simply by multiplying it with the emitter-specific consumption value, assuming that all emitters are the same. This may not necessarily be the case in a real data center, as operators can choose different emitters (especially at different bit-rates) for different switches or servers; but for this study we chose the simpler assumption of identical emitters everywhere.

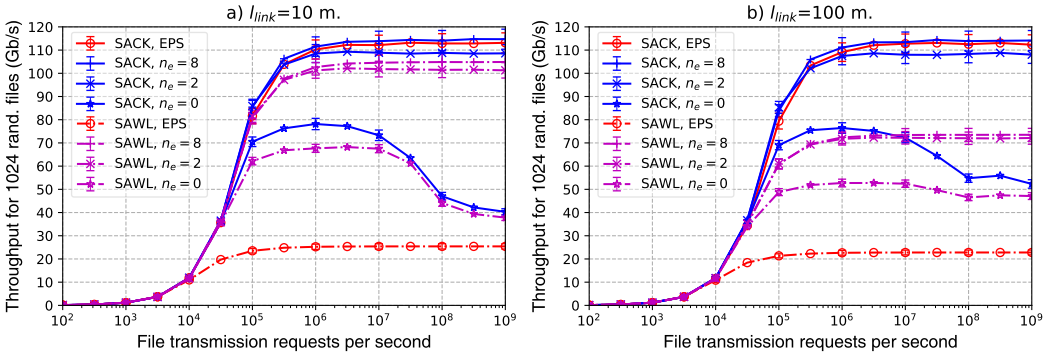


Figure 5.3 – Network throughput dependence on TCP CCA and switch type: a) $l_{link} = 10$ m, b) $l_{link} = 100$ m.

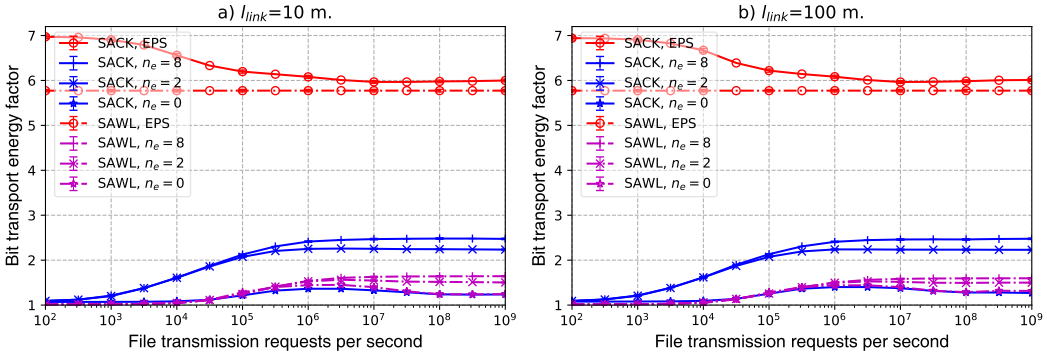


Figure 5.4 – Transmission energy cost dependence on TCP CCA and switch type: a) $l_{link} = 10$ m, b) $l_{link} = 100$ m.

Measurement of "bit transport energy factor" based on EO conversions suggests that optical links are active only when there is a packet to transmit. This condition is imposed by OPS and not always true for conventional EPS networks, as usually EPS maintains many point-to-point links that are always active for synchronization purposes. In order to make a fair energetic performance comparison of OPS and EPS networks, we consider the same conditions for both cases: links are active only when there is a packet to transmit, i.e. networks are asynchronous and use burst mode receivers and transmitters. In the scientific literature similar conditions were already considered for EPS point-to-point optical links with sleep mode [77] as the next step after IEEE 802.3az Energy Efficient Ethernet standard implementation. This assumption favors EPS networks in our results, as it limits the use of emitters and transmitters only to when data is being sent.

5.2 Energy Consumption on Switching in HOPS Data Center

5.2.1 Simulation Conditions

We assess the performance of the TCP SACK and TCP mSACK in the context of general conditions described in Sec. 3.5.

We consider the following values of buffer I/Os for switches: $n_e \in \{0, 2, 8\}$, where

$n_e = 0$ correspond to all-optical switch or OPS switch, while other values refers to hybrid switch and $n_e = 8$ corresponds to a fully-buffered switch (cf. Sec. 3.5.4). All-electronic switch, or EPS case, is as well reviewed.

For hybrid and all-optical switches, we use 1 ms as the initial value of Retransmission Time Out (RTO), the timer upon expiration of which a packet is considered lost and retransmitted), since it has proven to be favorable [31]. For the EPS case, we use 1 s for RTO initialization, as a relevant recommendation by IETF [69]: it limits unnecessary retransmissions (no packet contention) that would have been induced by 1 ms, still giving the same throughput.

File transmission is regulated by TCP Stop-And-Wait-Longer (SAWL) or TCP Selective ACK (SACK) flavor, which we found to be the two best-performing CCAs in terms of throughput for hybrid switches from the SAW and AIMD families in previous Ch. 4.

We follow a poissonian process of arrivals of new connection demands with mean rate of given file transmission requests per second between all of the servers, so as to study the performance in terms of throughput and "transmission energy cost" of a network with different switches and protocols under progressively increasing load.

To study how link length influences performance of CCA we consider the following values of link lengths in DCN: $l_{link} \in \{10, 100\}$. All links are bidirectional and of the same length, typical of DCs. We must note here, that it is the server-to-server path length that plays important role in TCP CCA performance, and not the composition of the path by links of same or different lenghts, thus the "same length" approach is valid in our case.

To reduce statistical fluctuations, we repeated every simulation 100 times with different seed for each set of n_e, TCP CCA and l_{link}. The mean throughput obtained is shown in Fig. 5.3 and transmission energy cost in Fig. 5.4, with 95% t-Student confidence intervals at every second point on the graph.

5.2.2 Evaluation of Results

For the case $l_{link} = 10$ m, as expected, EPS incurs the highest energy consumption by far, but gives almost the highest throughput using SACK, only edged out by hybrid switching with $n_e = 8$ and SACK, perhaps thanks to the cut-through nature of OPS. EPS with SAWL, which limits connections to one packet in flight, has a much lower throughput, almost ×4.5, for only a marginal energy gain. Taking EPS with SACK as a reference, at highest load, the transport-energy savings of optical and hybrid switching range between ×2.4 and ×4.8 (58–79%). For hybrid switches, the general result is that SACK gives the highest throughput but highest energy consumption, and vice-versa for SAWL, even with different values of n_e. Nevertheless, it's important to remark that SAWL with $n_e = 2$ loses only 10% of throughput to EPS and saves a factor ×4 (75%) energy-wise. If throughput is a priority, SACK with $n_e = 8$ is slightly better than EPS and still saves up to ×2.4 (58%) in transport energy.

For the case of $l_{link} = 100$ m at highest load, the energy consumption for different switches combined with CCAs is almost the same as with $l_{link} = 10$ m, but the throughput performance of SAWL and hybrid switches drops by 30%, which may make the energy savings less attractive. However, SACK gives the same throughput with hybrid and electronic switches both, allowing the same conclusion: it is still possible to save up to ×2.4 (or 58%) in transport energy consumption without losing network throughput. The drop of throughput in the case with SAWL and its absence in the case with SACK are explained by the fact that SAWL exploits link capacity less efficiently with its only one

unacknowledged packet in flight, contrary to SACK with several possible packets in flight. In general, SACK with EPS decreases the energy consumption with load increase: it's explained by features of SACK, with small latencies and in absence of losses capacity of network may be overestimated, leading to congestion and thus retransmissions, adding up to a higher energy consumption.

For the general case of the all-optical switch with $n_e = 0$, we notice that the throughput decreases by $\{60-70\}$% at highest load compared to the case of $n_e = 2$, without gaining much in energy consumption.

5.3 Conclusion

In this chapter we showed how introducing hybrid switches in DC networks related to real world cases (e.g. Facebook [71]) can decrease transport energy consumption at least by ×2 compared to electronic switches, while maintaining the same throughput. It was shown that this factor could be doubled up to ×4 while losing only 10% in throughput, thus letting us claim the beneficial character of DC network migration from EPS towards OPS on hybrid switches. DCs can benefit from hybrid switches that have a lower energy consumption than electronic, and a higher throughput and robustness than all-optical ones, using just a few electric ports and introducing specially-designed TCP protocols.

Chapter 6

Latency in DC Network on Different Switching Mechanisms

The Packet Switching (PS) technology is the essential enabling mechanism in nowadays telecommunications, including not only the Internet but also Inter and Intra Data Center (DC) communications. Core benefits of packet switching, such as statistical multiplexing and efficient capacity use, help to keep up with ever-increasing demands of high network throughput and "tail"(99th percentile) low latencies [78], especially in DCs [37]. Current PS technologies also ensure a possibility to meet the need for low-latency: a low cost and efficient solution Data Center TCP (DCTCP) CCA [37] recently was proposed. It successfully combines high network throughput and low latency altogether and is strongly encouraged to be deployed in DCs [79]. DCTCP is seen as the best solution for a low latency demand in Data Center Networks (DCN) employing Electronic PS technology.

In previous chapters, we combined two approaches of hardware-level (hybrid switch) and network-level (TCP CCA) and studied overall Hybrid OPS (HOPS) implemented in a DC. We introduced an adapted for HOPS DC network Stop-And-Wait-Longer (SAWL) CCA and reviewed the performance of a conventional Selective ACKnowledgement (SACK) CCA. We have found that HOPS offers the best throughput in a DC network, surpassing OPS and EPS. However, the important question of latency was not fully explored.

In this chapter, we introduce our analysis of the latency observed in HOPS DCs: we are reviewing not only TCP SAWL and TCP SACK, but also DCTCP, as a flagship TCP in the battle for low latency in DCs. As a latency measurement, we consider a Round Trip Time (RTT), i.e. time needed for a packet to be acknowledged. Also, we are introducing the measurement of Flow Completion Time (FCT), a metric that is related to latency and considered to be the most important for network state characterization [80].

6.1 DCTCP basics

6.1.1 Explicit Congestion Notification

In the context of Sender/Receiver computer communication, DCTCP relies on a mechanism, called Explicit Congestion Notification (ECN) [81] that involves switches that can indirectly notify Sender about congestion in the network. Such mechanism

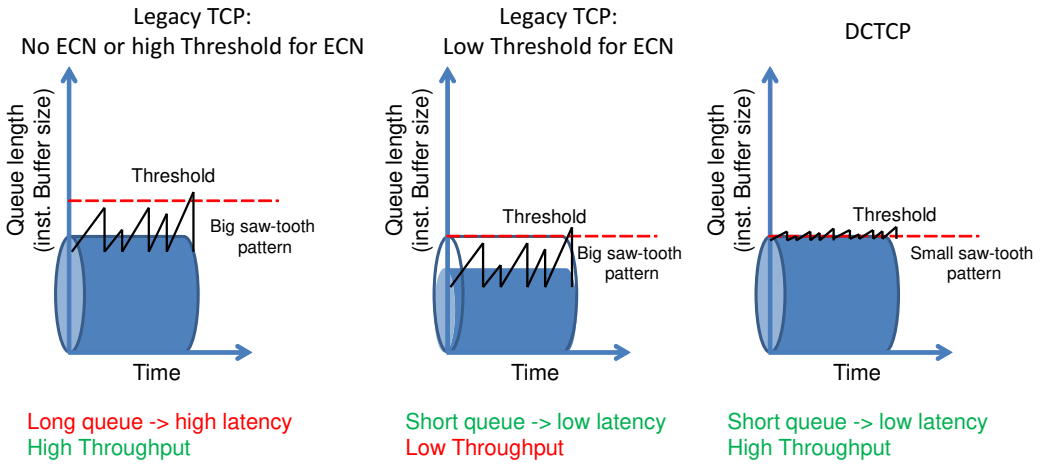


Figure 6.1 – Difference on instantaneous buffer size (queue length) with different TCP

consists of marking packets as Congestion Encountered (CE) by switches or routers in case the queue length (instantaneous buffer size) exceeds a predefined threshold – the k value. When CE packets get acknowledged by Receiver, the acknowledgment packets are attributed ECN Echo (ECE) mark. Upon reception by Sender of acknowledgment packets with ECE, the Sender can get informed about congestion on the way and can adapt its transmission characteristics, such as reduce Congestion Window (CWND). ECN was proposed to introduce intelligent management of congestion, so the buffer will not be always filled up, contributing to high latency (the bigger the queue length or instantaneous buffer, the longer packet stays in it). ECN mechanism, in turn, relies on classical TCP CCAs, such as TCP New Reno [82] or TCP SACK [72], just introducing additional conditions on CWND handling in case of ECE mark reception.

ECN notification initially was introduced to notify the sender about the congestion and reduce the CWND by half [81] on the occurrence of notification:

$$\text{CWND} = \frac{\text{CWND}}{2} \tag{6.1}$$

This approach is deemed to be aggressive in reducing the congestion window that does not take into account the level of congestion, just its existence. This resulted in exclusive behavior of latency or throughput: if one set buffer threshold too high [1] then the throughput is high, but latency is not low, if one sets buffer threshold low, one can achieve low latency, but also low throughput. The evolution of queue length of the buffer in both of these cases is represented in Fig. 6.1, inspired from [83]. When using just classical ECN mechanism we have a big "saw-tooth" pattern on queue length, which does not allow achieving simultaneously high throughput and low latency.

6.1.2 DCTCP

The contribution of DCTCP in comparison with ECN lies in an intelligent reaction to the ECN notification: it does not reduce CWND by half, but rather reduces CWND by extent of the congestion, i.e. in dependence from level of congestion in network. This

1. If it is never passed then it means absence of ECN and use of just pure TCP CCA.

extent is represented by a parameter $alpha$, so now the CWND is adapted accordingly to that factor [38]:

$$\text{CWND} = \text{CWND} \times \left(1 - \frac{alpha}{2}\right) \tag{6.2}$$

The parameter $alpha$ is evaluated when the "observation window end" variable is updated, and depends on a predefined weight g and on the ratio of bytes acknowledged with ECE to bytes acknowledged in general. More specifically, this ratio is defined by bytes sent by Sender and then acknowledged by packets marked with ECE to bytes sent by Sender and acknowledged in general over "observation window" period:

$$alpha = alpha \times (1-g) + g \times \frac{\text{Bytes Acknowledged with ECE}}{\text{Bytes Acknowledged}} \tag{6.3}$$

The "observation window end" variable is updated each time Sender receives an acknowledgement of packet with a sequence number exceeding current "observation window end". New "observation window end" is set to the sequence number of the packet that is going to be sent when Sender will receive next acknowledgement. Such sequence number is labeled as SND.NXT in conventional TCP. The "observation window" is defined by period of time between two consecutive "observation window end" updates. DCTCP does not react to congestion indications more than once for every window of data. [38]

According to [37] and [38] one can opt for use of delayed acknowledgments, i.e. Receiver emit one acknowledgement per one or two packets received, with the exception of the change of CE mark on packets received, i.e. if CE marks change from 1 to 0 or from 0 to 1. We regulate the use of delayed acknowledgment with variable m, which is equal to the number of DATA packets per one Acknowledgment packet.

The mechanism of DCTCP with its pseudo-code on Sender, Receiver and Switch side is represented in Fig. 6.2 and adapted from [83]: Sender sends packets, Switch marks them with CE mark in case of exceeding threshold of queue length or instantaneous buffer size, Receiver marks acknowledgments with ECE mark if CE packets were received. Upon reception of Acknowledgments by Sender, the new estimate of $alpha$ is calculated and then in case of reception of ECE mark CWND is updated accordingly to Eq. 6.2.

DCTCP lets us manage the queue length of the buffer more efficiently than just ECN mechanism, as shown in Fig. 6.1, with a small saw-tooth pattern. DCTCP let us work on the edge of the capacity: to have a high throughput and to have low latency, so needed in Data Center Networks.

The example of comparison of CWND evolution with and without DCTCP is provided in Fig. 6.3 and taken from [84]: the CWND evolution is not that aggressive and introduction of DCTCP let to have a more stable CWND, meaning stable high throughput and low latency.

6.2 Study of DCTCP performance in HOPS and OPS DCN

In this section, we study DCTCP applied to HOPS, EPS and OPS networks in order to understand how its parameters influence network performance and to choose the most adapted ones.

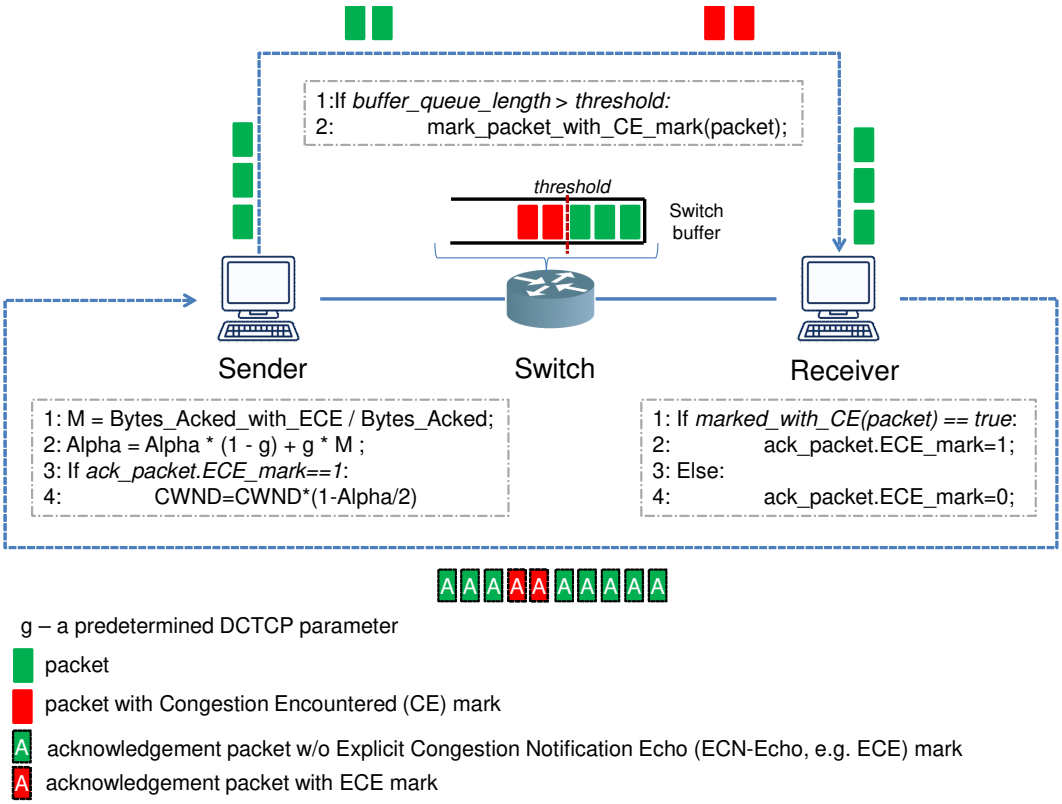


Figure 6.2 – DCTCP working principle

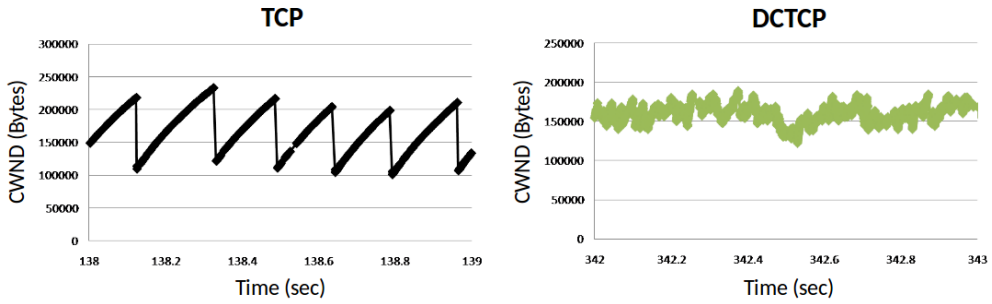


Figure 6.3 – Example of CWND evolution on DCTCP, taken from [84]

6.2.1 Simulation Conditions

We assess the performance of the DCTCP in the context of general conditions described in Sec. 3.5.

We study Data Center Network activity under DCTCP with different parameters in terms of throughput, 99th percentile of FCT and 99th percentile of RTT. We consider link length of $l_{link} \in \{10, 100\}$ m and next set of DCTCP parameters: $g \in \{0, 0.06, 0.12\}$; buffer threshold $k \in \{9064, 18128, 27192\}$ $Bytes$ corresponding to 1,2 and 3 data packets respectively; number of data packets per acknowledgement $m \in \{2, 1\}$,i.e. cases with and without delayed acknowledgments. We consider the following types of network: OPS with no buffer inputs/outputs $n_e = 0$, HOPS with $n_e = 2$ buffer inputs/outputs,

HOPS with with $n_e = 8$ buffer inputs/outputs, EPS switch. We consider a common $RTO_{init} = 1\ ms$ for all the cases. We already saw previously that there is no difference in network performance for EPS networks between cases $RTO_{init} = 1\ ms$ or conventional $RTO_{init} = 1\ s$, thus we are opting for $RTO_{init} = 1\ ms$ for EPS as well. Our choice is reinforced by the reported fact that the reduction of RTO can have a positive effect on EPS [79] as well.

Simulation results are a quite large set of data, which we providing in Appendix A. In this chapter we provide highlights of such data and present restricted set of parameters: $g \in \{0, 0.06\}$, $k \in \{9064, 27192\}\ B$, $m \in \{2, 1\}$ as DCTCP parameters; $l_{link} = 10$ m as for link length; OPS ($n_e = 0$), HOPS ($n_e = 8$) and EPS as network packet switching mechanisms. Conclusions on the influence of parameters l_{link} and n_e are made thanks to figures presented in Appendix A.

6.2.2 Throughput Study

We present on Fig. 6.4 average throughput with with 95% t-Student confidence intervals on every second point. We present here results obtained with a reduced set of parameters, however conclusions are made based on results on the whole set of parameters.

The k parameter, representing the threshold when the switch buffer starts marking packets, influences differently on different networks. In general, it does not influence on throughput of OPS and HOPS networks. OPS network does not have a buffer, so no packets marked, rendering DCTCP obsolete for OPS. HOPS network has a very low buffer occupancy, even when using conventional TCP CCAs, which is reviewed in Sec. 6.2.5, and confirmed in previous Ch. 5 with energy consumption review, i.e. a small number of packets pass through the buffer. This means that buffer is not occupied very often, and that is why k parameter doesn't change throughput (which is not true in case of 99th percentile of RTT, cf. Sec. 6.2.3). At the same time, k parameter does influence the throughput of the EPS network: the bigger k threshold, the bigger the network throughput. This is explained by the fact that in EPS switches buffers are used all the time, and if we are going to force them to have very low queue length, then links won't be used in full capacity. That is why to have better throughput it is recommended not to set k too low.

The g parameter follows the same behavior as k parameter: it is not influential for OPS and HOPS networks, but increasing such parameter makes better throughput of the EPS network. We suspect that is because of the same reason: low buffer occupancy does not give us the chance to exploit the full benefits of DCTCP. On a side-note, the case $g = 0$ corresponds to classic ECN mechanism, i.e. Eq. 6.2 turns into Eq. 6.1.

The m parameter makes throughput less performant if increased from $m = 1$ towards $m = 2$. We suspect that this comes from the fact that Jumbo Ethernet frames of 9064 B are too big to use them to transmit "mice" flows and at the same time to have one acknowledgment per $m = 2$ data packets. Some flows have size even smaller than standard Jumbo frame, implying only one data packet. This might produce a situation when Sender has finished transmission of a packet and is waiting for the acknowledgement, and at the same time Receiver is waiting for the next data packet, which will eventually arrive, but in form of re-transmission of first and only one packet.

The l_{link} negatively influences throughput while increasing only when we use a small buffer threshold, e.g. $k = 9064\ B$ or one Jumbo Frame: in that case DCTCP starts

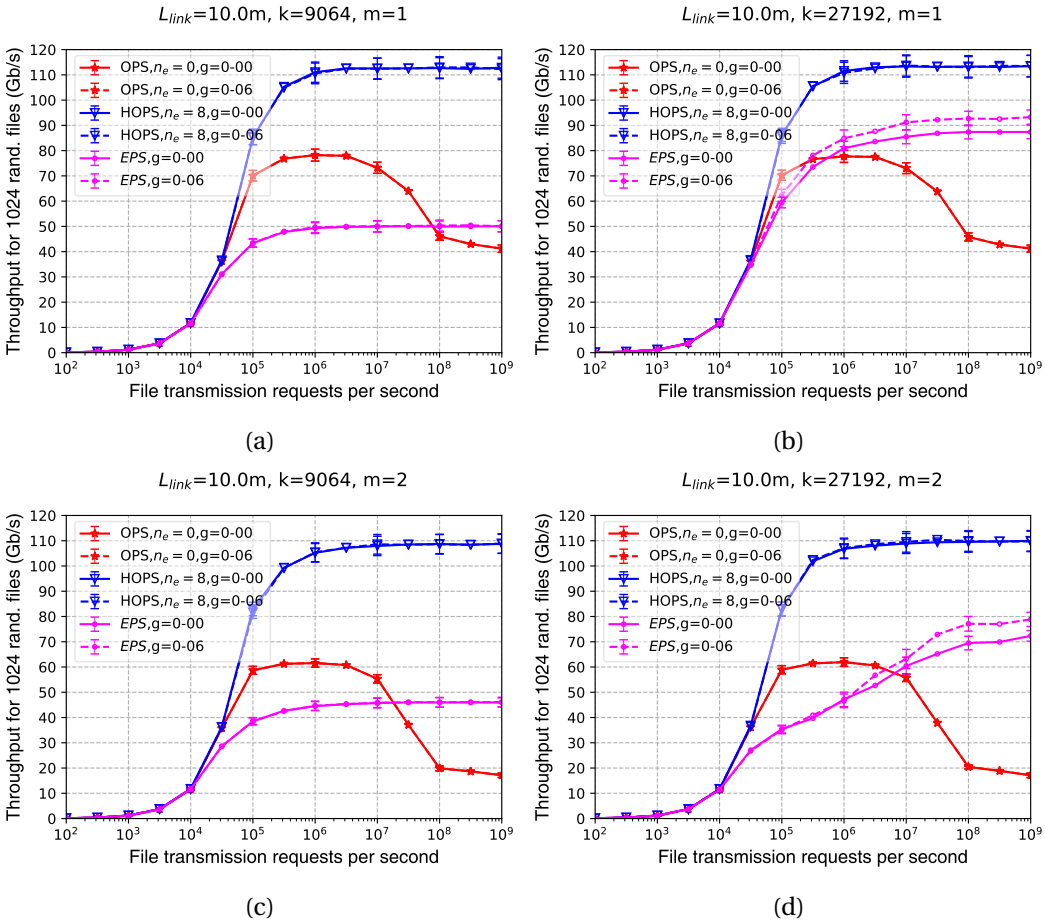


Figure 6.4 – Throughput dependence on DCTCP parameters: a) $k = 9064\ B$, $m = 1$, b) $k = 27192\ B$, $m = 1$ c) $k = 9064\ B$, $m = 2$, d) $k = 27192\ B$, $m = 2$

behaving like SAW, it does permit only one packet in buffer, preventing from physically filling a link completely when distances increases. The bigger k gets, the fewer changes occur in throughput. This is true for all types of networks considered.

In general, we conclude that the HOPS network performs better than both EPS and OPS networks, even with small buffer inputs/outputs $n_e = 2$.

6.2.3 99th percentile of RTT Study

We present in Fig. 6.5 the evolution of 99th percentile of RTT dependent on the load on the network: we provide these results on the reduced set of parameters, however, conclusions are made considering the complete set of them. Conclusions on the influence of parameters l_{link} and n_e are made based from full set of parameters considered (cf. Appendix A). We rely on RTT as on most important indicator of network latency, as it is independent of flow size, contrary to FCT.

The k parameter is the most important parameter influencing 99th percentile of RTT and it is true for all types of network that relies on buffer use (i.e. with exception OPS, where it has no power). In general, the bigger the threshold, the bigger the 99th percentile. This consequence is logical, as DCTCP is set to regulate buffer occupancy,

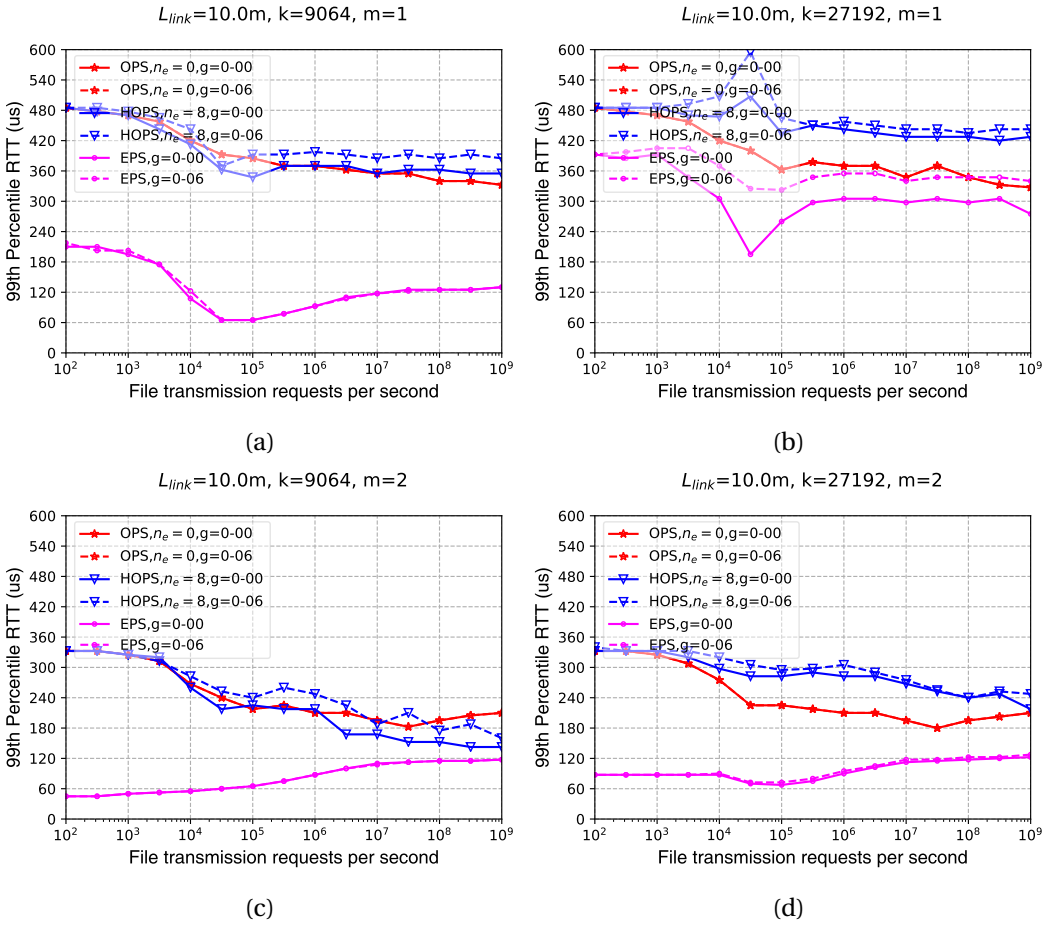


Figure 6.5 – 99th percentile of RTT dependence on DCTCP parameters: a) $k = 9064$ B, $m = 1$, b) $k = 27192$ B, $m = 1$ c) $k = 9064$ B, $m = 2$, d) $k = 27192$ B, $m = 2$

and if we keep the buffer almost always empty, packets will spend less time in the buffer, resulting in a small latency. This also serves as proof that DCTCP is implemented correctly in our simulator.

The g parameter follows the k parameter's behavior the same way as with throughput: the bigger g, the bigger the 99th percentile of RTT. The case of $g = 0$ represents a classical ECN mechanism, where CWND is halved each time, contributing to the lowest latency possible. If we are to increase g then we introduce less aggressive handling of CWND, increasing RTT (however still reduced compared to conventional TCP CCAs, cf. Sec. 6.3).

The m parameter can reduce the 99th percentile of RTT if we introduce delayed acknowledgement, e.g. $m = 2$. This might seems beneficial, but it comes with the cost of decreased throughput. Additionally, it appears that with $m = 2$ the changes of the parameter g do not affect RTT performance at all, which is not true if $m = 1$ is used.

In general, surprisingly, it is discovered that EPS networks under DCTCP have the lowest 99th percentile of RTT than OPS or HOPS, making DCTCP as a very powerful tool for EPS networks. However, first, we will show that it is not true for the 99th percentile of FCT, and second, in Sec. 6.3 we will show that combination HOPS+SAWL has better 99th percentile of RTT compared to EPS+DCTCP.

It appears that l_{link} does not influence on 99th percentile of RTT. This might be

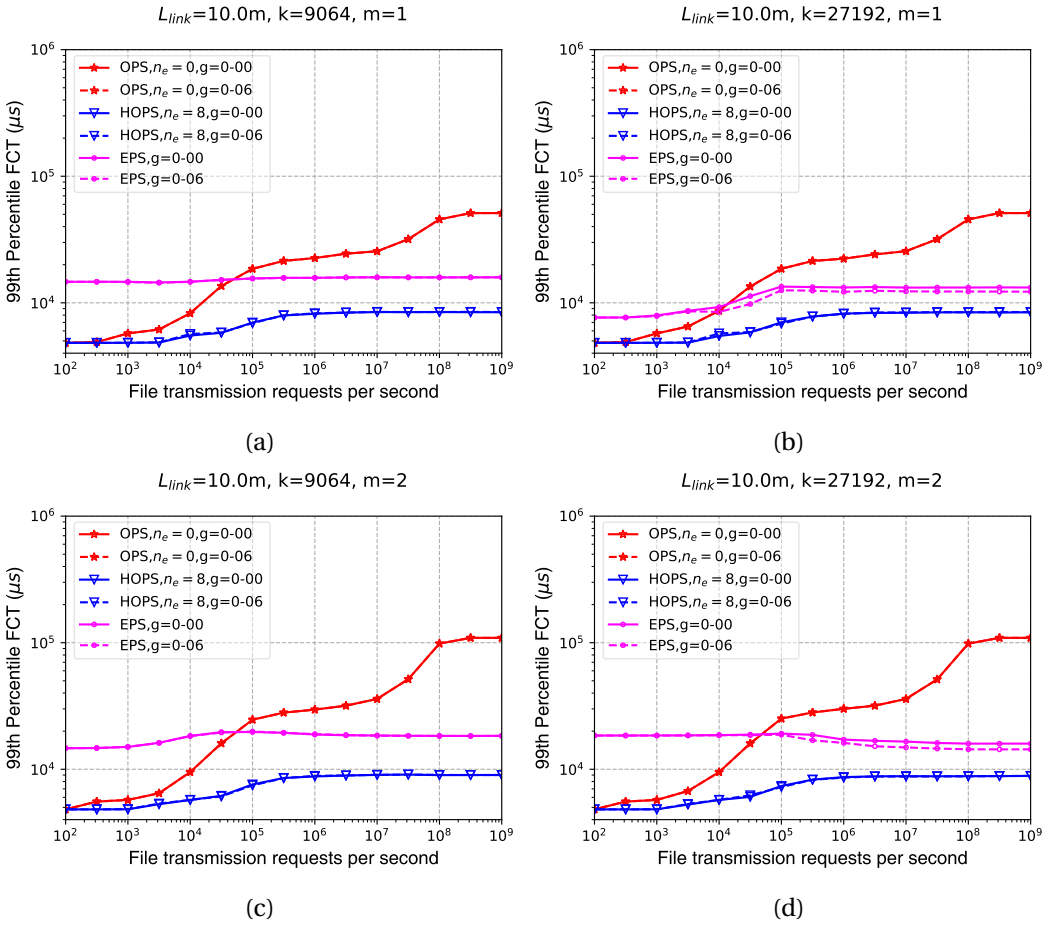


Figure 6.6 – 99th percentile of FCT dependence on DCTCP parameters: a) $k = 9064$ B, $m = 1$, b) $k = 27192$ B, $m = 1$ c) $k = 9064$ B, $m = 2$, d) $k = 27192$ B, $m = 2$

seem counter-intuitive as when we increase l_{link} we increase propagation time, but is explained by the fact that propagation delay increase by order of µs [2], while 99th percentile is on the order or hundreds of µs. Additionally, the 99th percentile of RTT represents "the worst" latency possible, where other factors dominate, such as network load and congestion.

6.2.4 99th percentile of FCT Study

We present 99th percentile of FCT changes in Fig. 6.6. This parameter would help us to evaluate additional to RTT latency aspect in the network that employs DCTCP. FCT is defined as the time needed for a file or flow to be transmitted and completely acknowledged, thus 99th percentile of FCT is the "worst" case scenario of how long it will take to transmit a file in a network. Conclusions on the influence of parameters l_{link} and n_e are made based from full set of parameters considered (cf. Appendix A).

The k parameter influences only EPS networks, if we consider only 99th percentile: the bigger the buffer threshold, the smaller is the 99th percentile of FCT. However, this

2. The biggest propagation delay for $l_{link} = 10$ m in considered DCN is 0.29 µs and for $l_{link} = 100$ m it is 2.9 µs.

relation does not hold every time, as we will see in Sec. 6.3 if the buffer threshold is infinite (i.e. TCP SACK) the 99th percentile of FCT is worse. HOPS and OPS networks performance does not change.

The g parameter follows k behavior in the same way as in the case of throughput and RTT, not influencing HOPS and OPS network, and following rule the bigger g parameter, the less is the 99th percentile.

The m parameter makes 99th percentile of FCT higher, if increased from $m = 1$ to $m = 2$, influencing all types of network considered.

The l_{link} does not influence much performance metrics of FCT, apparently for the same reasons explained previously for the 99th percentile of RTT.

In general, the HOPS network performs better than EPS, which, in turn, performs better than OPS: HOPS outperforms OPS almost by a factor of 10 in general. We bring attention to the fact that such a conclusion holds even if we consider HOPS with $n_e = 2$ buffer inputs/outputs instead of $n_e = 8$: adding just 2 buffer inputs/outputs to all-optical switch let us boost the performance.

6.2.5 Buffer Occupancy

Even though it seems that k and g parameters do not strongly influence the performance of HOPS network, this is not true: if we compare performance of HOPS under DCTCP with performance under TCP SACK we will notice: 1) difference in buffer occupancy of the switch, and 2) consequent difference in RTT and FCT (reviewed further in Sec. 6.3).

In this subsection, we review the influence of DCTCP on buffer occupancy of hybrid (HOPS, $n_e = 8$) and all-electronic (EPS) switches. We consider the transmission of the same predefined batch of 1024 files on the load of 10^9 requests per second. As a result, we measure the amount of time the buffer held a certain amount of data, in terms of percent of total transmission time needed to transmit 1024 files in a network with $l_{link} = 10$ m. This measurement is presented in 3 categories for different levels of switches: L1, L2, and L3 (as in topology presented in Ch. 3). Consequently, this metric is averaged among 32 L1 Switches, 32 L2 Switches, and 16 L3 Switches.

Measurements in the form of Cumulative Distribution Function (CDF) for HOPS network is provided in Fig. 6.7 and measurements for EPS network is provided in Fig. 6.8. For each network we consider next two sets of DCTCP parameters: $k = 2719200\ B$, $m = 1$, $g = 0$ and $k = 27192\ B$, $m = 1$, $g = 0.06$. The first case is aimed at an approximation of TCP SACK (no weight to g and high buffer threshold) and the second is aimed at representation of DCTCP.

On Fig. 6.7 we notice first that, for HOPS, the buffer is empty almost all the time no matter what CCA we use. More than 70 % of transmission time buffer is empty no matter the level of the switch. Second, we notice that L1 switches are the most loaded in the system. Third, we notice that DCTCP helps us to unload the buffers of L1 switches and move 99th percentile from 960784 B towards 81576 B, i.e. reduce it more than by a factor of 10.

On Fig. 6.8 we present DCTCP influence on EPS network. In comparison to HOPS, the buffers are empty only 40 % of the time, also meaning idle time (switch is not used for data transmission). The fact that L1 switches are loaded the most, holds as well as in the HOPS network. Same way DCTCP helps to unload buffers of such switches: moving 99th percentile from 607288 B towards 99704 B.

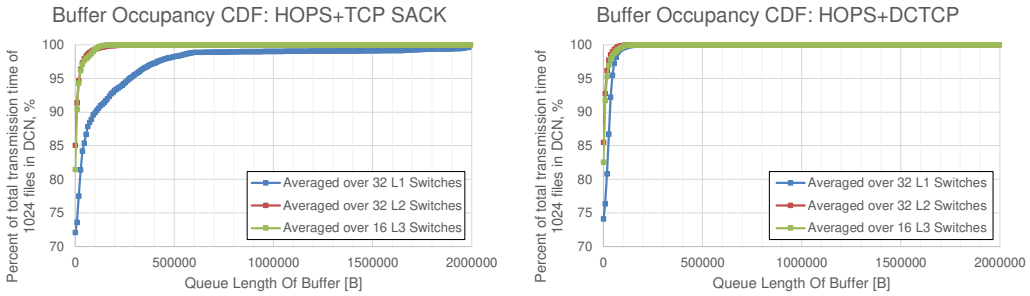


Figure 6.7 – Buffer Occupancy of switches of different levels to transmit same set of 1024 files arriving at 10^9 req./s: left) HOPS, $n_e = 8$ over TCP SACK ($k = 2719200\ B, m = 1, g = 0.00$), right) HOPS, $n_e = 8$ over DCTCP ($k = 27192\ B, m = 1, g = 0.06$)

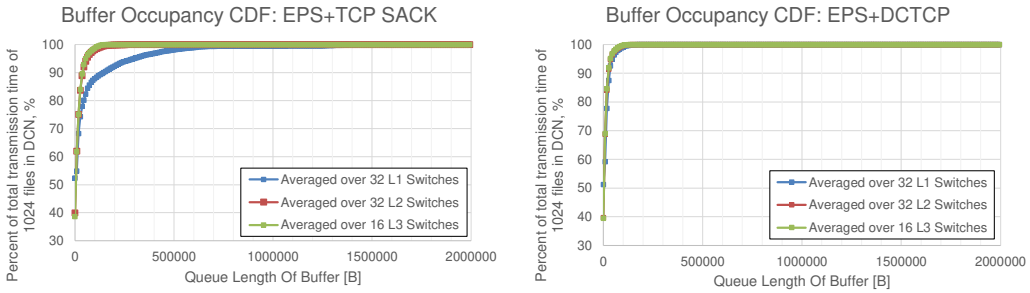


Figure 6.8 – Buffer Occupancy of switches of different levels to transmit same set of 1024 files arriving at 10^9 req./s: left) EPS, $n_e = 8$ over TCP SACK ($k = 2719200\ B, m = 1, g = 0.00$), right) EPS, $n_e = 8$ over DCTCP ($k = 27192\ B, m = 1, g = 0.06$)

While considering the TCP SACK case, we notice the particular characteristic of buffer occupancy of L1 switches: for HOPS, switches are empty almost all the time, while for EPS it is only half of the time, but HOPS has bigger 99th percentile. We explain that by possible reordering of packets emitted in one CWND. In HOPS one packet might be put into the buffer, i.e. switched in store-and-forward mode, while others are switched optically, i.e. in a cut-through mode. In EPS network reordering should be less present as all packets pass by buffer and switched in the same store-and-forward mode, however, it is not eliminated, as different network paths are used to transmit the same flow due to per-packet load-balancing.

While considering DCTCP, we don't observe particularities observed for TCP SACK. In HOPS network 99th percentile is better, or on the same level as in EPS. Furthermore, in HOPS L1 switches buffers are almost empty all the time, while in EPS they are only around 30 % of the time.

In general, we observe that the application of DCTCP strongly influences buffer occupancy the same way as in EPS networks.

6.2.6 Conclusion on DCTCP parameters choice

DCTCP shows itself as an efficient CCA to reduce latencies in DCN. It is most influential on EPS networks, less on HOPS networks. We can regulate the level of throughput,

RTT, and FCT for the EPS network, but only RTT for the HOPS network. However, it is considered as advantageous as HOPS exhibits the best throughput and FCT if compared to EPS.

DCTCP possesses a set of parameters: buffer threshold k to mark packets, weight g to regulate CWND and m to regulate the number of packets to acknowledge with one acknowledgment. Each of these parameters influences in its own way the network performance in different aspects. For instance, in the case of EPS a small threshold of k and zero g reduces the 99th percentile of RTT, but decreases throughput at the same time. That is why the choice of these parameters is a trade-off: one must choose parameters that make the network performance acceptable from all sides. That is why we advise on the next set of parameters: $k = 27192\ B$ as a buffer threshold, $g = 0.06$ as a weight in CWND reduction, $m = 1$ meaning absence of delayed acknowledgment. This set of parameters gives a decent level of throughput and 99th percentile of FCT while exhibiting low 99th percentile of RTT for the EPS network. As we said before, throughput and 99th percentile of FCT in HOPS doesn't strongly depend on the choice of k and g considered in this section, that is why we use further these parameters also for HOPS network.

6.3 Latencies of different CCAs achievable in DC

In this section, we are evaluating latencies achieved in general in HOPS, EPS and OPS networks and compare DCTCP performance with TCP SACK and TCP SAWL.

6.3.1 Simulation Conditions

We assess the performance of the network in the context of general conditions described in Sec. 3.5.

We consider the following values of buffer I/O for switches: $n_e \in \{0, 8\}$, where $n_e = 0$ correspond to all-optical switch or OPS switch, while other values refers to hybrid switch and $n_e = 8$ corresponds to a fully-buffered switch (cf. subsec. 3.5.4). All-electronic switch, or EPS case, is also reviewed. Further, in this study we will refer to case $n_e = 0$ as OPS network, the case with $n_e = 8$ as HOPS network. We consider $n_e = 8$ because we want to have a fair comparison of HOPS with EPS, with no contention factor influencing the result.

All links are bidirectional and of the same length $l_{link} = 10\ m$, typical of DCs. We follow a Poissonian process of arrivals of new connection demands with mean rate of given file transmission requests per second between all of the servers. We study the network performance in terms of network throughput, average and 99th percentile of RTT, average and 99th percentile of FCT, under progressively increasing load.

File transmission is regulated by either: *i)* TCP SAWL (Ch. 4), *ii)* or TCP SACK (Ch. 4), *iii)* or by DCTCP [37]. SAWL and SACK were already seen as two best-performing CCAs in terms of throughput in HOPS. We consider DCTCP as a possible solution to decrease latencies in DC. For DCTCP we have chosen the next parameters for implementation: $k = 27192$ B, $m = 1$ and $g = 0.06$, other parameter variants are considered less or equally performant in this study conditions, so we do not review them here. We used TCP SACK as an underlying CCA for DCTCP implementation. We use 1 ms as the initialization value of RTO timer not only for OPS and HOPS cases but as well for EPS, since the reduction of RTO can have a positive effect on EPS [79] as well.

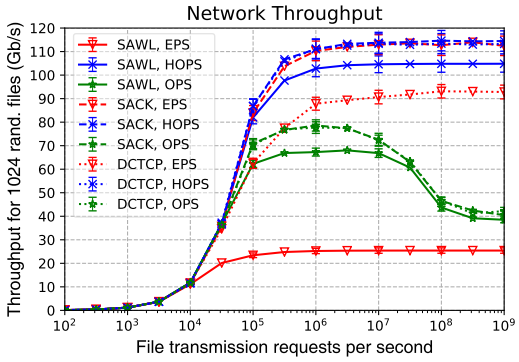


Figure 6.9 – Throughput dependence on CCA and load

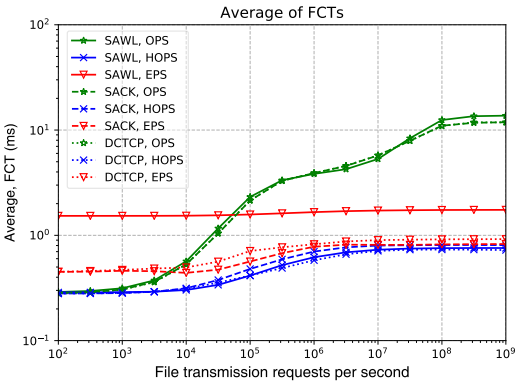


Figure 6.10 – Average FCT in a DCN

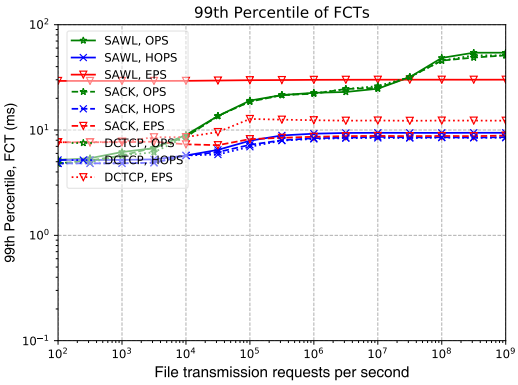


Figure 6.11 – 99th percentile FCT in a DCN

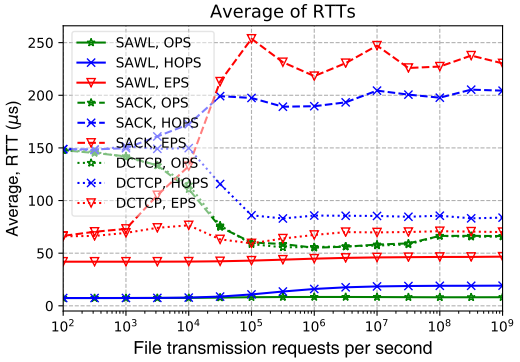


Figure 6.12 – Average RTT in a DCN

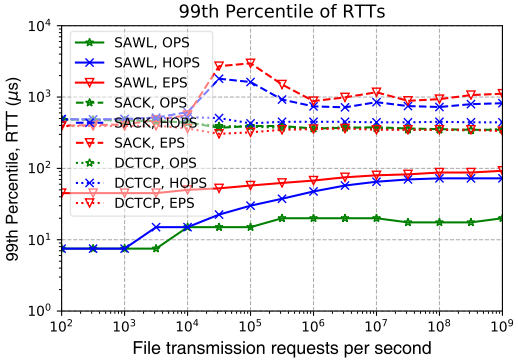


Figure 6.13 – 99th percentile RTT in a DCN

6.3.2 Performance Analysis

The mean throughput obtained is shown in Fig. 6.9 with 95% t-Student confidence intervals at every second graph point. Latency related characteristics, such as average FCT, 99th percentile of FCT, average RTT and 99th percentile of RTT presented on Fig. 6.10, Fig. 6.11, Fig. 6.12 and Fig. 6.13 respectively.

TCP SACK, conventional CCA, is seen as unadapted to deliver low latencies both for EPS and HOPS. Both yield close results and are well above those from other protocols:

99th percentiles of RTTs ≈1ms, when in other cases they are well below. Still, one could notice that SACK delivers better performance under HOPS than under EPS both in RTTs statistics and throughput under high load. OPS case under SACK is far from optimal; it has low throughput and drastically raises FCT during network load increase compared to EPS or HOPS.

TCP DCTCP is a logical successor of the TCP SACK in our case, as it is built on top of it and brings the best from conventional CCAs. While passing from SACK to DCTCP the network under HOPS does not lose in throughput but decreases average RTT by ×2.5 and 99th percentile by ×1.8. Also, DCTCP under HOPS delivers the best FCT among other types of networks. Despite this, DCTCP seems to influence EPS performance more noticeably: average RTT and 99 percentile are better than those of HOPS (by around 15μs and 150μs respectively), however at the cost of throughput. Passing from SACK to DCTCP, EPS loses around 20% of its throughput and still underperforms in terms of the FCT. OPS's performance doesn't differ from SACK, as OPS doesn't have buffers, on which DCTCP relies on.

TCP SAWL, being developed specifically for HOPS, remains one of the best candidates among CCAs and brings out the best from HOPS. HOPS under SAWL yields average RTT not bigger than 20μs and 99th percentile below 75μs; unachievable neither by HOPS under DCTCP nor by EPS in general. The FCT is almost the same as those in the case of conventional TCP CCA (SACK or DCTCP). Only one minor drawback comes from the throughput that is lower than in HOPS under DCTCP only by 10%. At the same time, one should note that EPS under DCTCP loses to SAWL under HOPS not only in throughput, but also in latencies, and FCT. While considering OPS, SAWL delivers best RTTs, but drastically loses in FCT and throughput both to EPS and HOPS.

Measurement of the 99th percentile of FCT presented in Fig. 6.11. FCT is an important parameter on the same role as the 99th percentile of RTT, however, it is not representative the same way as it depends on flow size. Short flows would have low FCT, while large flows would have large FCT. Thus, measuring the 99th percentile of FCT we would be heavily depended on flow size. At the same time, the 99th percentile of RTT is independent of flow size. Nevertheless, we provide 99th percentile of FCT in this study as well to be compared with average FCT in Fig. 6.10. The 99th percentile of FCTs mostly follows the characters of lines of average FCT, mostly with a difference of the factor of 10, except for the OPS case on high load, with its only 4 times difference. This may be explained by the hypothesis that the main contributors to the 99th percentile of FCT are the following: time of packet spent in the buffer, amount of retransmissions and flow size. At low loads for HOPS and OPS only flow size contributes to 99th percentile, whereas at high loads for OPS case it is mostly retransmissions, and for HOPS it is a flow size and time of packet spent in the buffer. The OPS's 99th FCT percentile stays the biggest among reviewed solutions.

6.4 Conclusions

In this chapter, we have reviewed the performance of DCTCP in the context of HOPS, EPS and OPS networks and determined parameters that are beneficial for latency reduction while maintaining throughput. Further, we studied latencies that could be achieved in DCN if applying not only DCTCP but also TCP SACK and TCP SAWL, showing that TCP SAWL is the most adapted CCA to be applied in HOPS DCN for $l_{link} = 10$ m.

This study shows how the introduction of hybrid switches in DC networks makes HOPS as the most attractive among PS technologies. In combination with SAWL, HOPS surpasses by far the EPS in latencies or throughput while being on the level with the throughput of HOPS under DCTCP. The combination with DCTCP helps considerably decrease latencies compared to SACK, without losing in throughput, which is the case for EPS.

Chapter 7

Hybrid and Optical Packet Switching Supporting Different Service Classes in DC Network

In previous chapters we analyzed the gain from use of the hybrid switch in a Data Center (DC) network by introducing Hybrid Optical Packet Switching (HOPS): we showed that HOPS with a custom designed TCP can outperform OPS and EPS in throughput. Furthermore, we have managed to show the possibility of 4 times reduction in DC energy consumption for data transport coming from OEO conversions while using HOPS compared to EPS. In this study we aim to investigate not only a combination of HOPS with custom design of TCP, but also the influence of the introduction of Classes of Service, i.e. switching and preemption rules for packets of different priorities.

Considering the general interest in the scientific and industrial communities to implement different packets priorities in Data Centers (DCs), as well as the problem of traffic isolation for tenants in DC [85], we implement the idea presented by Samoud et. al. [30] and investigate the benefits of application of such technology in a DC network. We successfully show that one can considerably improve the performance of network consisting of hybrid switches with a small number of buffer inputs for high priority connections while keeping it on a good level for default connections. Additionally, we show that high priority connections in OPS network also can profit from the introduction of classes of service, matching or even surpassing the performance of the network consisting of hybrid switches with a small number of buffer inputs without classes of service.

7.1 Class Specific Switching Rules in OPS and HOPS

7.1.1 Packets Preemption Policy

The difference of present study from previous ones is in fact that we use custom switching rules, that are specific for class of the optical packet. If before we didn't consider the preemption of the packet on its way to output of the switch or the buffer, now we do consider it.

The switching algorithm for a hybrid switch is adopted from [30] and implements different bufferization and preemption rules for different packets classes. We consider

Algorithm 1 Preemption Policies in a Hybrid Switch

```
1: procedure SWITCH (PACKET P)
2:     prio ← p.priority_class
3:     switch_out ← get_destination_azimuth(p)
4:     if switch_out.is_free() then                          ▷ General switching rule of HOPS
5:         switch_out.receive(p)
6:     else if buffer_in.is_free() then
7:         buffer_in.receive(p)
8:     else if prio==R and buffer_in.receiving(D) then       ▷ Try to buffer R & preempt D
9:         buffer_in.preempt_last_packet(D)
10:        buffer_in.receive(p)
11:    else if prio==R and switch_out.receiving(D) then      ▷ Try to switch R & preempt D
12:        switch_out.preempt_last_packet(D)
13:        switch_out.receive(p)
14:    else if prio==R and buffer_in.receiving(F̃) then       ▷ Try to buffer R & preempt F̃
15:        buffer_input.preempt_last_packet(F̃)
16:        buffer_input.receive(p)
17:    else if prio==R and switch_out.receiving(F̃) then      ▷ Try to switch R & preempt F̃
18:        switch_out.preempt_last_packet(F̃)
19:        switch_out.receive(p)
20:    else if prio==F̃ and switch_out.receiving(D) then      ▷ Try to switch F̃ & preempt D
21:        switch_out.preempt_last_packet(D)
22:        switch_out.receive(p)
23:    else
24:        drop(p)
```

three of them: Reliable (R), Fast (F) and Default packets (D). R packets are those to be saved by any means, even by preemption of F or D packets on their way to buffer or switch output. F packets could preempt only D packets on their way to the switch output. D packets cannot preempt other packets.

The priority distribution in the DC network is adopted from [30] and taken from the real study on core networks [86]. This may seem improper for DCs, however, we seek to study the performance of the hybrid switch in a known context. Also, it will be shown below that the distribution considered allows us to organize a pool of premium users (10%) of R connections in DCs that could profit from the best performance, while other users almost wouldn't be influenced by performance loss. F packets can preempt D packets only on the way to switch output, while R packets first would consider preemption of D packet being buffered. Thus F packets had lower delay than R packets [30]. However, further it will be shown that this device-level gain doesn't translate to network-level gain in a DC network in terms of Flow Completion Time (FCT), and R connections perform better than F. That's why here we refer to Fast (F) as Not-So-Fast ($\tilde{F}$) packets and connections. Eventually, in this study we consider, that 10% of connections have R priority, 40% of connections have $\tilde{F}$ priority, 50% of connections have D priority.

When a packet enters the switch it checks if required Azimuth output (i.e. switch output) is available. If yes, the packet occupies it. Otherwise, the packet checks if any of buffer inputs are available. If yes, it occupies one and starts bufferization. If none of the buffer inputs are available, in the case of absence of preemption policy in a switch the

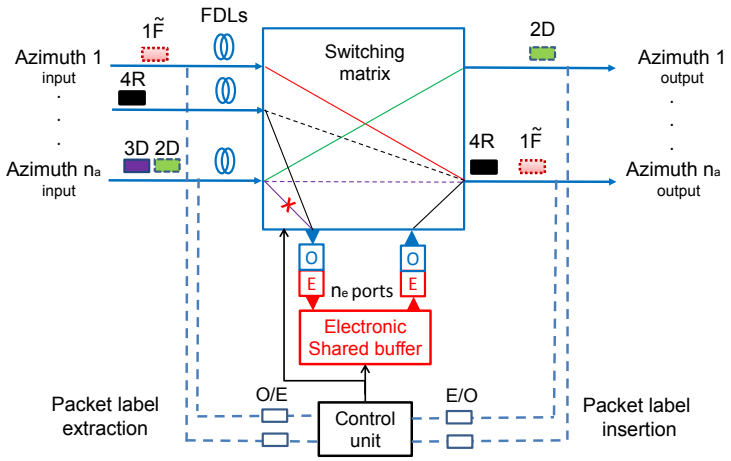


Figure 7.1 – Example of Class-specific Routing Rules for hybrid switch with $n_e = 1$.

packet would be simply dropped. Here, we consider a switch with preemption policy that would follow the steps of algorithm presented in Alg. 1. If a packet of any type is buffered, it is re-emitted FIFO, as soon as required switch output is available.

7.1.2 Manageability of Packet Loss

On a Fig. 7.1 we demonstrate the Alg. 1 on a hybrid switch with only one buffer input/output $n_e = 1$ for simplicity. We have packets: 1 F̃ – an Not-So-Fast packet, 2D and 3D – two Default packets, and 4R – an Reliable packets. Packet 2D requires an available output of the switch and is transmitted without any OEO conversions directly. Then, packet 1 F̃ arrive at the switch and is directed to the required output of the switch. After, a Default 3D packet arrives at the switch, and is redirected to the buffer, as the required Azimuth is occupied by the packet 1 F̃. Further, a Reliable packet 4R arrives and requires the same output as packets 1 F̃ and 3D, 1 F̃ is still occupying the switch output and 2D occupying sole buffer input. In an agnostic switching rules case the packet 4R would be lost, and packets 1 F̃ and 3D would be transmitted as shown in Fig. 3.2 in Sec. 3.2, but in the current case it is a Default packet is preempted and lost, while 4R packet is put in the buffer and then transmitted to the output of the switch.

In this context class specific switching rules help us to control, what packet would be lost, 4R or 3D. In other words this translates to the general rule: we can decide what packets would be lost, while keeping more or less same overall PLR. Class specific switching rules allow us to take control and making packets drop more manageable.

7.2 Study Conditions

As in our previous work, we simulate the communications of DC servers by means of optical packets. The general conditions described in Sec. 3.5. We study DC network performance for two groups of scenarios: DC with classes of service using preemption policy outlined in Sec. 7.1, and DC with switches that don't have any preemption rules. For each scenario we consider OPS and HOPS case.

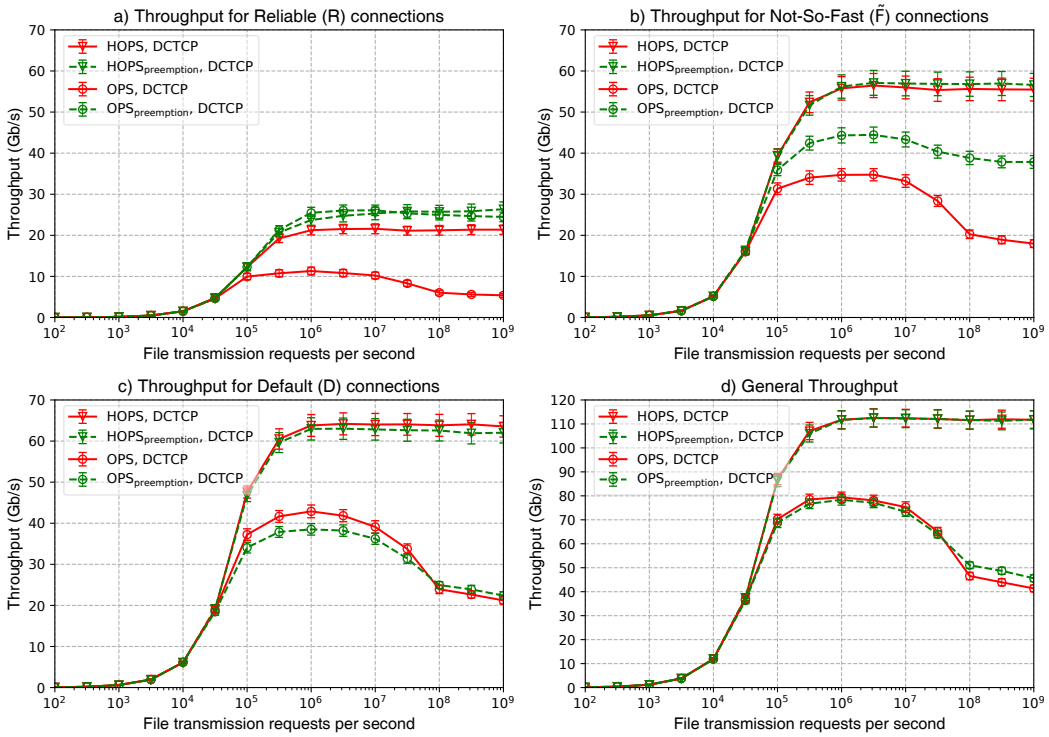


Figure 7.2 – DC network's throughput for connections: a) Reliable (R) connections, b) Not-So-Fast (F̃) connections, c) Default (D) connections, d) Overall Network Performance

The actual transmission of each data packet is regulated by the DCTCP CCA [38], developed for DCs, which decides whether to send the next packet or to retransmit a not-acknowledged one. CCA uses next constants: $k = 27192$ B, $m = 1$, $g = 0.06$, as favorable for HOPS. We apply the crucial reduction of the initialization value of RTO towards 1 ms, as advised in [31].

We further enhanced a previously developed discrete-event network simulator based on an earlier hybrid switch simulator [30], extended so as to support class-specific switching rules. The simulated network consists of hybrid switches supporting class-specific switching rules with the following architecture: each has n_a azimuths, representing the number of input/output optical ports, and n_e input/output ports to the electronic buffer, as shown in Fig. 7.1. The case of the bufferless all-optical switch (OPS) corresponds to $n_e = 0$, for the case of the hybrid switch (HOPS) we consider $n_e = 2$. In previous chapters we considered fully-buffered hybrid switches ($n_e = 8$) and all-electronic switches (EPS), but in this study we don't, as class-specific switching rules are considered as not needed as all of the packets can be saved from contention just by use of the buffers.

The network is characterized by the network throughput (in Gb/s) and average FCT (in μs) for each type of connections and general case as a function of the arrival rate of new connections, represented by the Poissonian process. We have chosen FCT as a metric considered to be the most important for network state characterization [80].

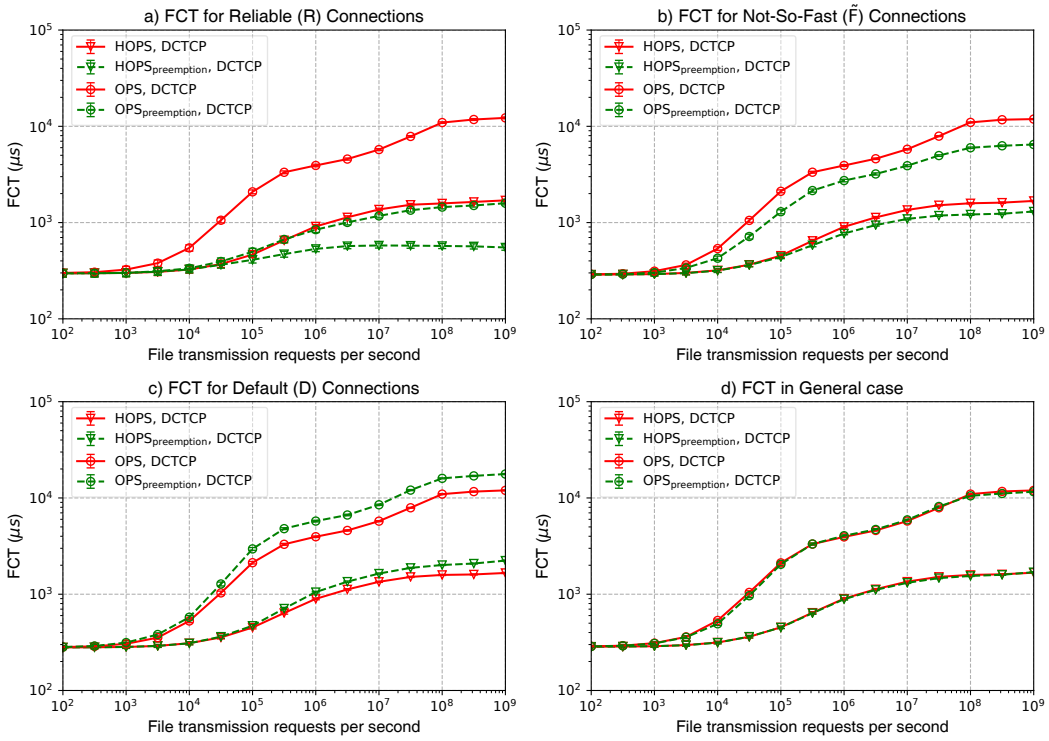


Figure 7.3 – DC network's Flow Completion Time for connections: a) Reliable (R) connections, b) Not-So-Fast (F̃) connections, c) Default (D) connections, d) Overall Network Performance

7.3 Advantages of using Class Specific Switching Rules

We present here the results of our study and their analysis. To reduce statistical fluctuations, we repeated every simulation a hundred times with different random seeds for $n_e = 0$ (OPS) and $n_e = 2$ (HOPS). The mean throughput and mean FCT are represented in Fig. 7.2 and in Fig. 7.3 with 95% t-Student confidence intervals, for three types of connections: R, F̃ and D connections. We take as a reference results from the network without packet preemption policy. The division of connections to classes is just corresponding to classes' percentage of connections in the network. We define high load as more than 10^5 connections per second.

While comparing just OPS and HOPS, it is seen that in general HOPS outperforms or has the same performance as OPS, but with the cost of only $n_e = 2$ buffer inputs.

R connections benefit the most from the introduction of the Classes of Service and preemption policy as it seen on Fig. 7.2a) and Fig. 7.3a) both in the cases of OPS and HOPS. Throughput for R connections in HOPS network rises by around 25% (Fig. 7.2a), while in OPS case it rises by a factor 2.5 at least on high load, matching the performance of HOPS network. We would like to bring readers attention on the fact that it seems to be low throughput, compared to other classes of service, but this is the mere effect of the fact that in the network only 10% of connections are of type R. However, if one considers the FCT, which is comparable with other types of classes and lowest among them, then the preemption policy's benefits are more evident. On the highest considered load OPS reduces its FCT almost by a factor of 8, while HOPS reduces it by at least a factor of 2,

keeping it on the level of tens of μs. Even if OPS's FCT doesn't match FCT in the case of HOPS while considering Classes of Service, it does match the FCT in the case of HOPS without Classes of Service. While applying preemption policy, connections are indeed Reliable. On Fig. 7.4 we can see that PLR (ratio of packets lost due to preemption or dropping to packets emitted by servers) decreases by around factor of 10, while for $\tilde{F}$ and D PLR remains around the same level (not shown here).

$\tilde{F}$ traffic benefits less than R traffic from introduction of Classes of Service, but the gain is still there. For OPS we managed to boost the throughput by almost 30-100% on the high load, while for HOPS the gain is less evident. However, when we consider FCT on Fig. 7.3b) we can see that OPS decreases its FCT by almost a factor of 2 for high load, and HOPS around 25%. HOPS FCT for $\tilde{F}$ packets is bigger than for those of reliable (R), contrary to what may be induced from [30], where they are labeled as Fast (F). This may be explained by the fact that the delay benefits for F packets are on the order of a μs, while here FCT is of an order of tens and hundreds of μs, and is defined mostly by TCP CCAs when contention problem is solved.

D traffic does not benefit from the introduction of Classes of Service, and it is on its account the gains for R and $\tilde{F}$ traffic exists. However, while considering the performance reductions, we notice almost unchanged throughput for HOPS case, and for OPS the drop of only 10% at most, which could be seen as a beneficial trade-off in R and $\tilde{F}$ traffic favor with their boost of performance both in throughput and FCT.

The network as a whole, regardless of the presence of Classes of Service, performs the same, which is expected, as connections occupy limited network resources. We can observe that the gain due to introduction of Classes of Service for R and $\tilde{F}$ traffic decreases with the increase of number of buffer inputs/outputs (i.e. from $n_e = 0$ towards $n_e = 2$), and for fully-buffered switch ($n_e = n_a = 8$) the gain would be 0, because no packet would ever require preemption, only bufferization. However, there are technological benefits to use small number of buffer input/outputs as it directly means simplification of switching matrix ($n_a = 8$, $n_e = 2$ means 10x10, $n_a = n_e = 8$ means 16x16 matrix) and reduction of number of burst receivers (inputs) and transmitters (outputs) for buffers. In the case of EPS, the gain would be also 0, but in general EPS entails an increase in energy consumption for OEO conversions compared to HOPS by a factor of 2 to 4 on high load, as we saw in Ch. 5.

While observing the network performance overall, we see that introduction of Classes of Service both in OPS and HOPS helps to boost the performance for the R and $\tilde{F}$ connections, while keeping the performance of D connections relatively on the same level. This fact could lead to economic benefits in a Data Center: charge more priority clients for extra performance, almost without loss of it for others. Furthermore, using pure OPS instead of HOPS in DCs may be economically viable, as OPS delivers the best possible performance to R connections, on the level of HOPS performance for $\tilde{F}$ connections, and relatively low performance for D connections, since high performance may be not needed for D connections.

7.4 Conclusions

In this study we enhanced the analysis of HOPS and OPS DC networks by applying classes of service in terms of preemption policy for packets in optical and hybrid switches, while solving the contention problem. In the case of HOPS we demonstrated that with

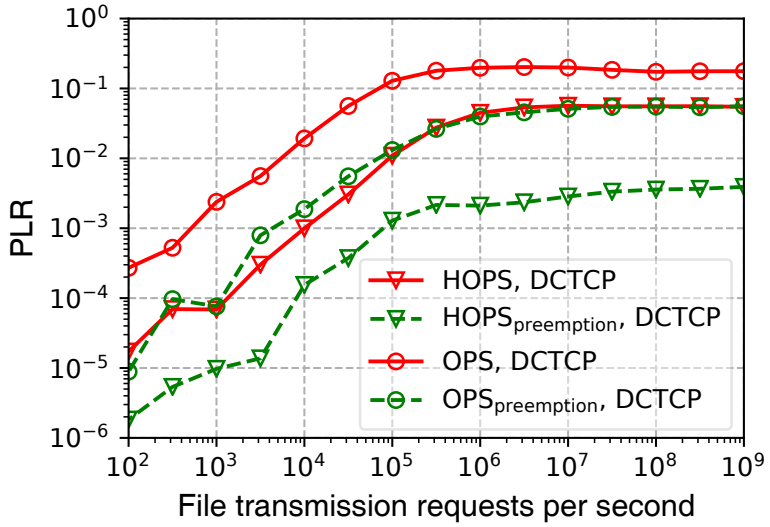


Figure 7.4 – Mean PLR of Reliable (R) Connections

custom packet preemption rules, one can improve the performance for Reliable and Not-So-Fast class connections, almost without losses for Default connections. Furthermore, we showed that Classes of Service can boost the performance of OPS for Reliable and Not-So-Fast class connections, match or bring it on the level of those in HOPS. This proves that OPS could be used in DCs, delivering high performance for certain connections, while Default class connections are still served on an adequate level.

It remains to be seen whether these results remain with a different service class distribution; and whether an actual low-latency service class can be implemented (e.g. using another TCP CCA).

Chapter 8

Conclusions and Future Work

8.1 Summary and Conclusions

The Data Center networks could benefit from hybrid switches that have lower energy consumption than electric ones, and a higher throughput and robustness than all-optical ones, using just a few electric ports and introducing the specially designed TCP protocols. This could be applied to any type of networks and load by changing the number of the electric ports and carefully adjusting the TCP algorithm.

Taking into account that the solutions for constructing the hybrid switch already exist, the crucial components as switching matrices and burst receivers are already a reality; considering the topology of the network, that is already in place (Facebook data center); the Hybrid Optical Packet Switching (HOPS) enabled by TCP Congestion Control Algorithms (CCAs) could be implemented in reality. The possibility of turning HOPS into reality is justified by the benefits of HOPS compared to existing Electronic Packet Switching (EPS). Throughput of HOPS not only matches or even surpass the throughput of EPS, but also entail two others crucial benefits: such as reduction of the energy per bit (reduction of Optical-Electronic-Optical (OEO) conversions) and latency (cut-through mode of operation for most of the packets).

In this work we extensively studied Data Center Networks (DCN) in the context of application of different switching mechanism and different TCP CCAs. Primarily we followed next three axes of DCN performance investigation: Throughput, Energy Consumption and Latency.

We started our investigation relying on measurement of DCN throughput under combination of switching mechanisms and TCP CCA used. We reviewed already developed TCP CCAs, such as TCP Stop-And-Wait (SAW) for Optical Packet Switching (OPS) and TCP Selective ACKnowledgment (SACK) for EPS networks. When applying these CCAs in context of HOPS networks we successfully showed that HOPS networks outperforms both OPS and EPS networks. Using insights on HOPS functioning we came up with new TCP CCA: TCP Stop-And-Wait-Longer (SAWL). TCP SAWL leverages throughput given by TCP SAW under HOPS and let us to fully benefit from electronic shared buffer. Even if TCP SACK still delivers best throughput under EPS and HOPS networks, TCP SAWL under HOPS shows very close to SACK performance but delivers outstanding performance in latency and energy consumption, compared to SACK.

We continued our investigation with measurement of energy consumption in DCN, trying to quantify OEO conversions reduction by OPS ans HOPS networks compared

to EPS networks. Our key findings show that HOPS consumes at worst by factor of 2 less energy than EPS networks at their best. We can bring that factor towards 4 by using combination of TCP SAWL + HOPS, trading off just several percents from best throughput possible.

We conducted complex study on latency in DCNs. We introduced to our analysis DCTCP, a flagship TCP currently applied in EPS DCNs in pursuit for minimum latency. We have found that combination DCTCP + HOPS can decrease latency of scheme SACK + HOPS and outperform DCTCP + EPS in FCT and Throughput, but cannot compete with combination TCP SAWL + HOPS, that delivers the best FCT and RTT, again showing benefits of application of HOPS in general and with TCP SAWL in particular in DCN.

Additionally we have investigated the question of traffic management in DCN if to introduce different Classes of Service (CoS) onto HOPS and OPS data centers. Our studies have shown that one can improve the performance for Reliable (R) class connections, almost without losing it for Default connections, by applying special packet switching rules in a hybrid switch.

8.2 Future Research Directions

Our investigation till now didn't touch several important questions, such as application of Wavelength Division Multiplexing (WDM) and study of Wide Area Network (WAN) topologies. Future studies can be conducted in order to answer them.

The study of influence of WDM technology on OPS network comes coupled with the study of WAN topologies, since, currently, WAN networks are inseparable from WDM technologies. As well WAN topologies would complete our study of application HOPS on different type of networks.

Additional studies on All-Optical Wavelength Converters (AO-WC) can be conducted in context of HOPS WDM networks, as they can leverage Wavelength Continuity constraint of current WDM network and introduce packet switching's statistical multiplexing gain.

Till the moment we considered only homogeneous networks, i.e. composed of only of one type of switch. It is necessary to investigate the concept of joint use of electronic and hybrid switches in order to study the interest in progressive integration and replacement of electronic switches with hybrid in real standard fat-tree DC networks and how this will influence throughput and energy consumption.

The experimental work in the laboratory was not envisioned at the beginning of this research project, however, the opportunities to do it are actively sought. That experimental work tied up with HOPS would allow a deeper evaluation of the HOPS properties and characteristics.

New and unexplored technology aspects of HOPS do not limit possible the scope of future works. There is also place for TCP CCA development and improvements.

One can further study TCP SAWL and determine conditions on when "learning",i.e. dynamic adjustment of the p parameter may be justified. Additionally there might be some improvements in using TCP mSACK for underlying DCTCP algorithm.

Appendix A

DCTCP parameters influence on Network Performance

In this appendix we provide results of simulation of Data Center Network activity under DCTCP with different parameters in terms of Throughput on Fig. A.1 and Fig. A.2; 99th Percentile of FCT on Fig. A.5 and Fig. A.6; 99th Percentile of RTT on Fig. A.3 and Fig. A.4. We consider link length of $l_{link} \in \{10, 100\}$ m and next set of DCTCP parameters: $g \in \{0, 0.06, 0.12\}$; buffer threshold $k \in \{9064, 18128, 27192\}$ $Bytes$ corresponding to 1,2 and 3 data packets respectively; number of data packets per acknowledgement $m \in \{1, 2\}$.

On each graph we provide simulation of next type of networks: OPS with no buffer inputs/outputs $n_e = 0$, HOPS with with $n_e = 2$ buffer inputs/outputs, HOPS with with $n_e = 8$ buffer inputs/outputs, EPS switch. We consider a common $RTO_{init} = 1$ ms for all the cases. We consider class-agnostic switching rules without packets preemption. All other parameters are defined in Ch. 3.

Analysis and conclusion on these results are provided in Ch. 6, Sec. 6.1.

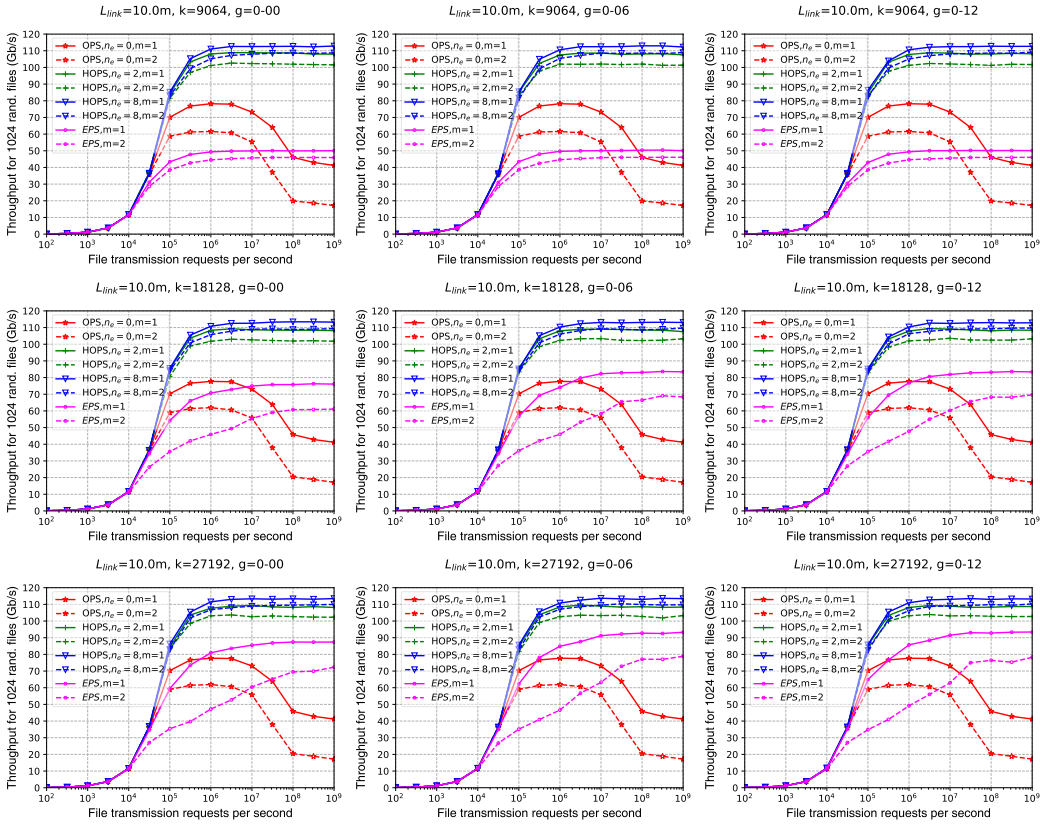


Figure A.1 – DCN ($l_{link} = 10m$) Average Throughput dependence on DCTCP parameters

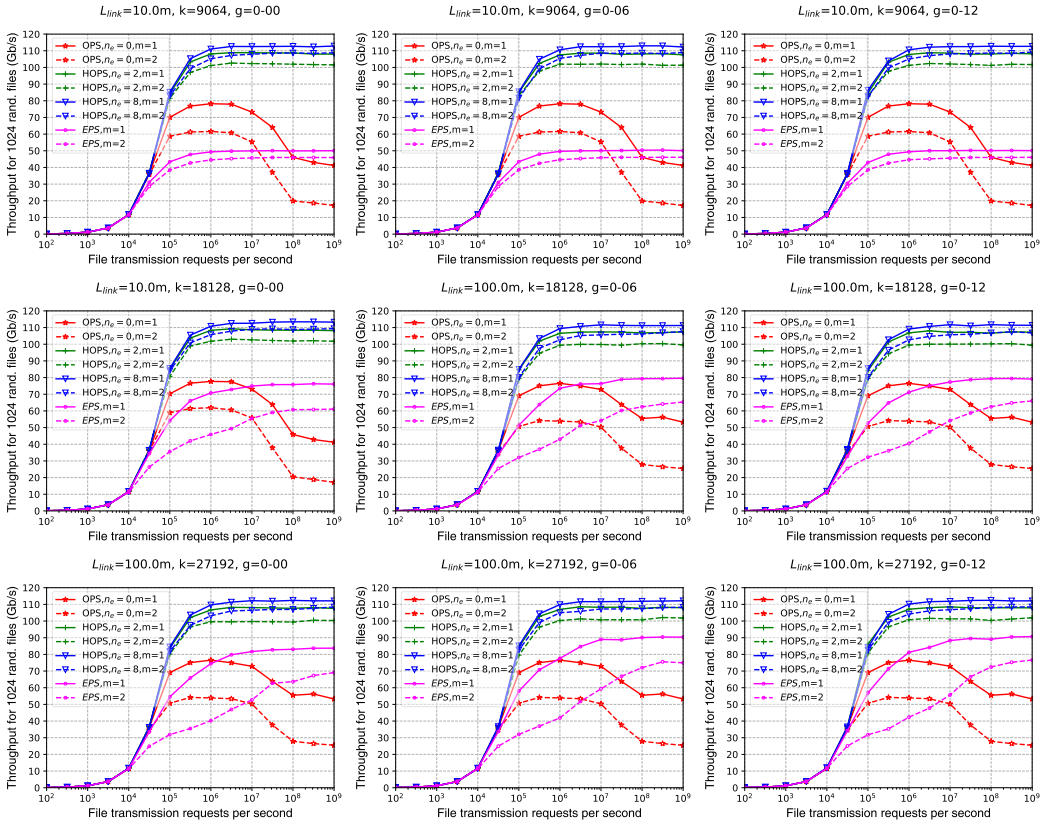


Figure A.2 – DCN ($l_{link} = 100m$) Average Throughput dependence on DCTCP parameters

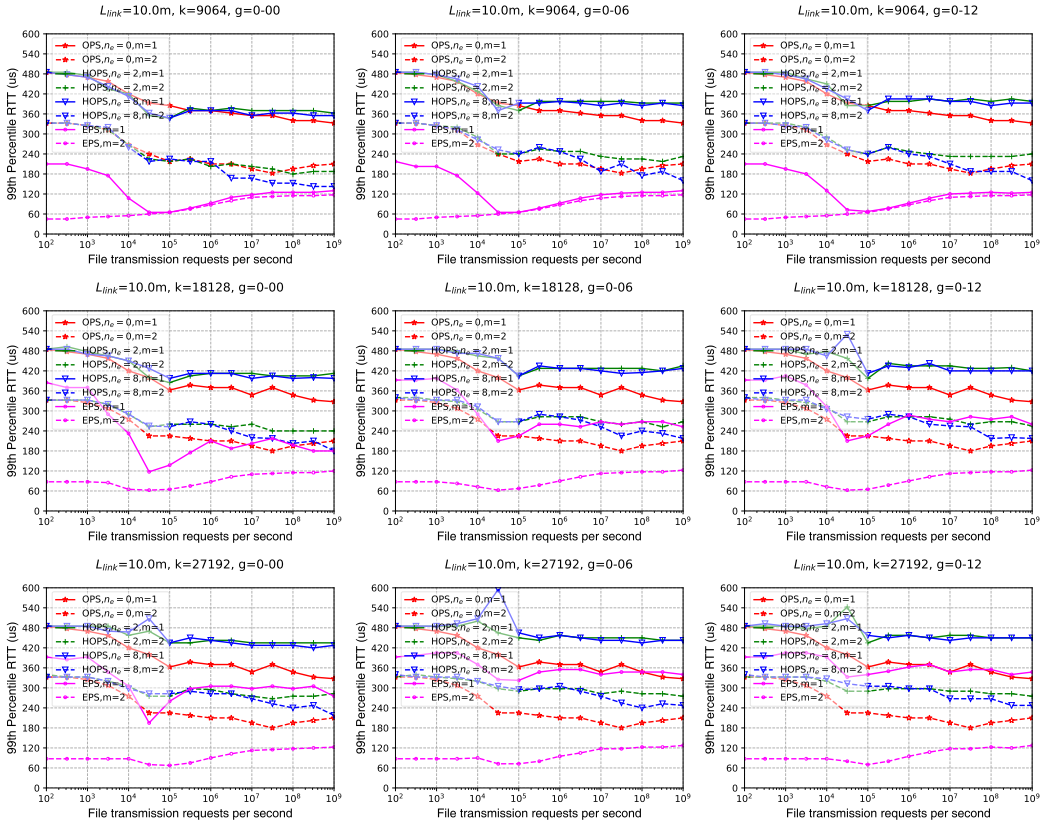


Figure A.3 – DCN ($l_{link} = 10m$) RTT 99th Percentile dependence on DCTCP parameters

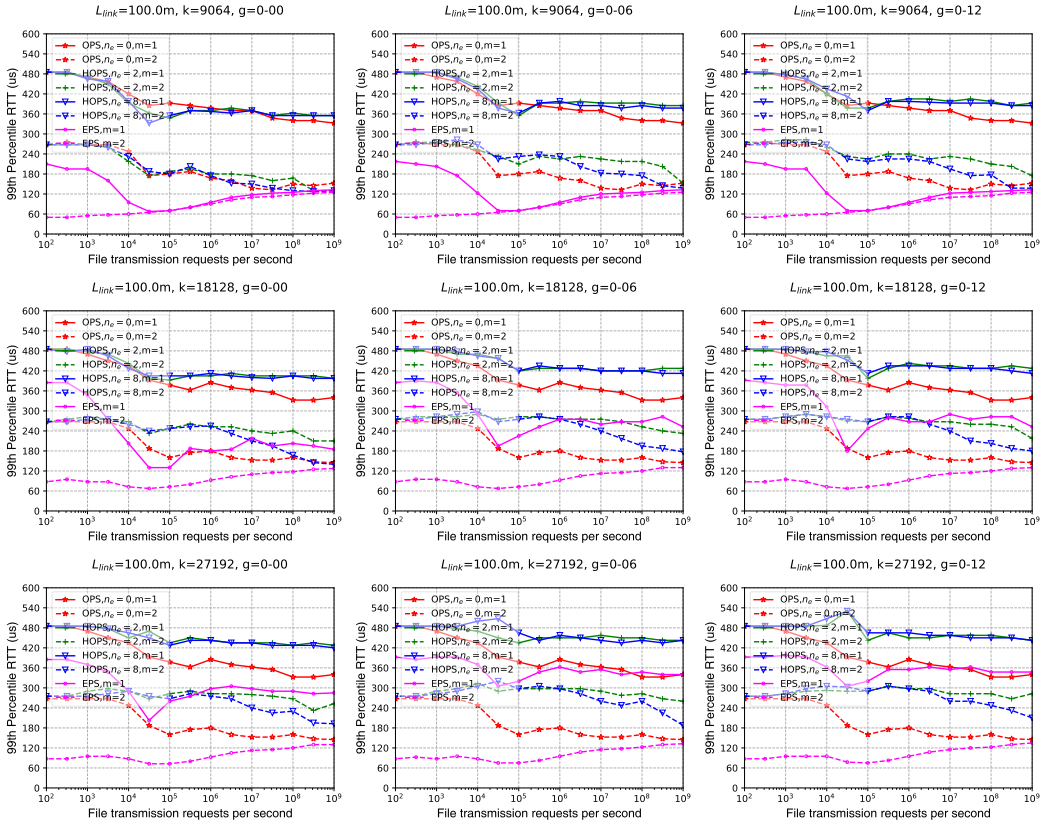


Figure A.4 – DCN ($l_{link} = 100m$) RTT 99th Percentile dependence on DCTCP parameters

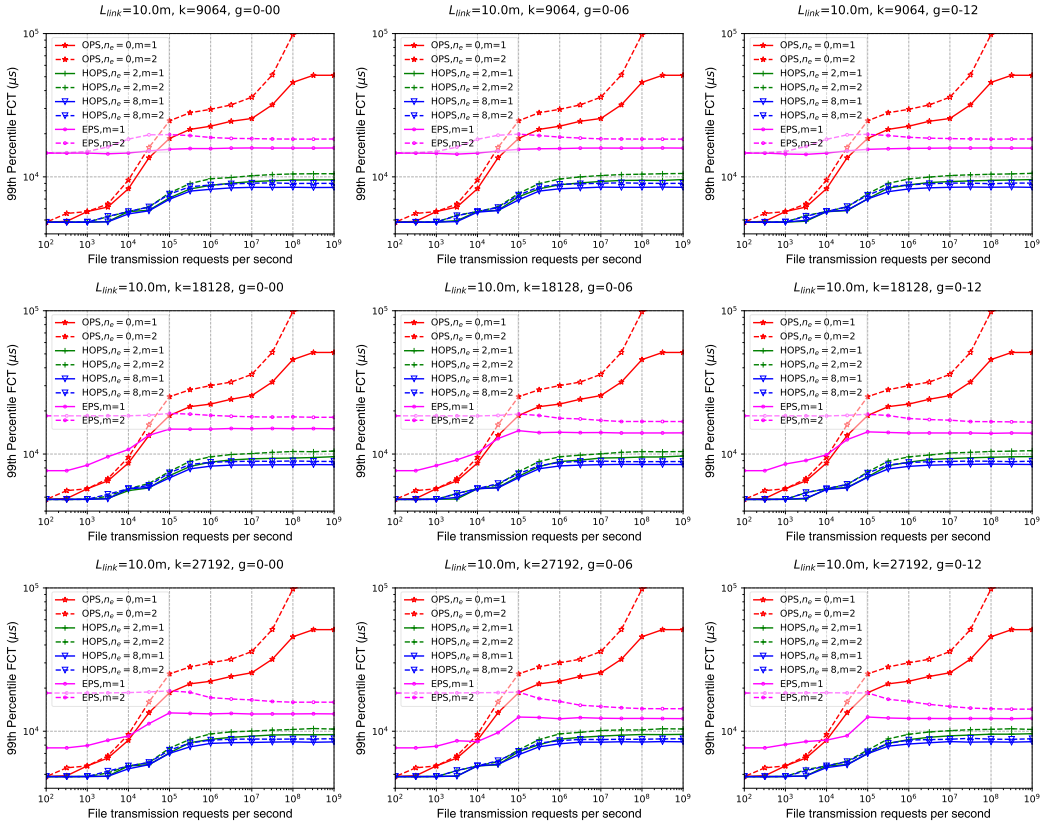


Figure A.5 – DCN ($l_{link} = 10m$) FCT 99th Percentile dependence on DCTCP parameters

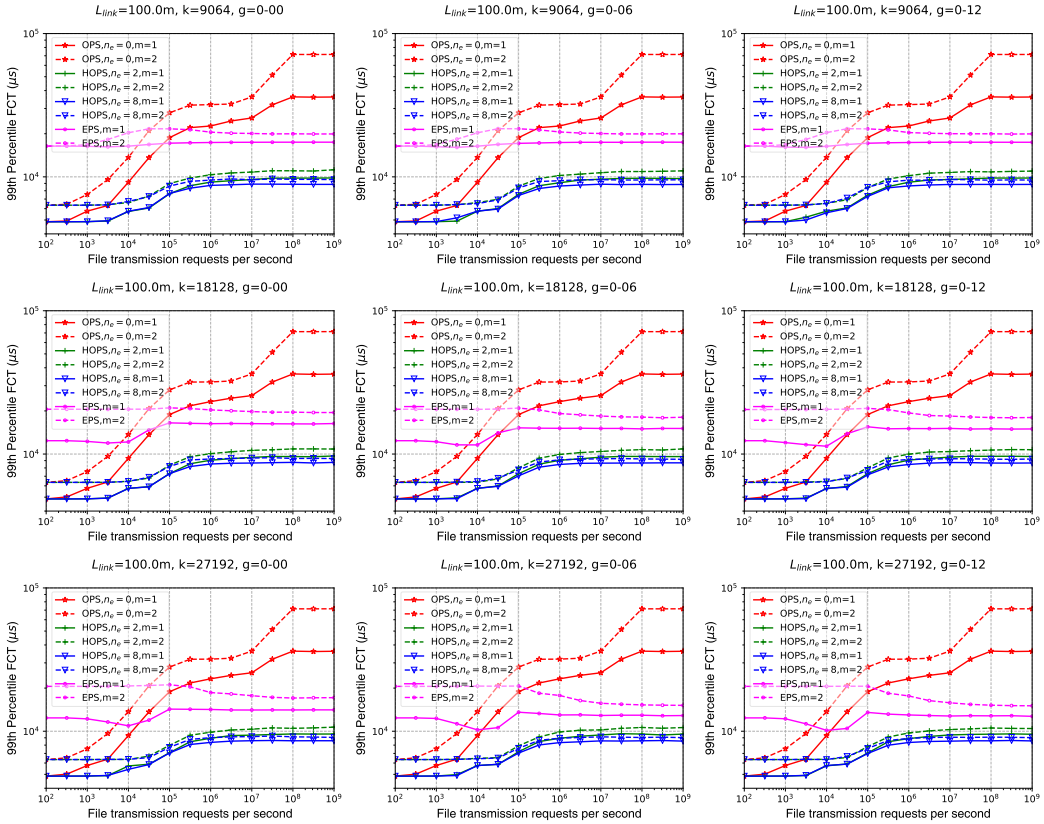


Figure A.6 – DCN ($l_{link} = 100m$) FCT 99th Percentile dependence on DCTCP parameters

Bibliography

[1] Cisco, INC, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022." Online: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html, 2018. Accessed: 2019-08-07.

[2] Finisar, INC, "400GBASE-LR8 QSFP-DD Optical Transceiver." Online: https://www.finisar.com/optical-transceivers/ftcd1323e1pcl, 2019. Accessed: 2019-08-07.

[3] Nokia, INC, "Alcatel-Lucent Submarine Networks and Nokia Bell Labs achieve 65 Terabit-per-second transmission record for transoceanic cable systems." Online: https://www.nokia.com/about-us/news/releases/2016/10/12/alcatel-lucent-submarine-networks-and-nokia-bell-labs-achieve-65-terabit-per-second-transmission-record-for-transoceanic-cable-systems/, 2016. Accessed: 2019-08-07.

[4] ETSI, "Mobile technologies - 5g, 5g specs: Future technology." Online: https://www.etsi.org/technologies/5g. Accessed: 2019-08-07.

[5] X. Liu and F. Effenberger, "Emerging optical access network technologies for 5g wireless [invited]," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 8, pp. B70–B79, December 2016.

[6] I. A. Alimi, A. L. Teixeira, and P. P. Monteiro, "Toward an efficient c-ran optical fronthaul for the future networks: A tutorial on technologies, requirements, challenges, and solutions," *IEEE Communications Surveys Tutorials*, vol. 20, pp. 708–769, Firstquarter 2018.

[7] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," in *OSDI'04: Sixth Symposium on Operating System Design and Implementation*, (San Francisco, CA), pp. 137–150, 2004.

[8] S. M. Rumble, D. Ongaro, R. Stutsman, M. Rosenblum, and J. K. Ousterhout, "It's time for low latency," in *Proceedings of the 13th USENIX Conference on Hot Topics in Operating Systems*, HotOS'13, (Berkeley, CA, USA), pp. 11–11, USENIX Association, 2011.

[9] R. Kapoor, G. Porter, M. Tewari, G. M. Voelker, and A. Vahdat, "Chronos: Predictable low latency for data center applications," in *Proceedings of the Third ACM Symposium on Cloud Computing*, SoCC '12, (New York, NY, USA), pp. 9:1–9:14, ACM, 2012.

[10] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," *SIGCOMM Comput. Commun. Rev.*, vol. 39, pp. 68–73, Dec. 2008.

[11] J. Perelló, S. Spadaro, S. Ricciardi, D. Careglio, S. Peng, R. Nejabati, G. Zervas, D. Simeonidou, A. Predieri, M. Biancani, H. J. S. Dorren, S. D. Lucente, J. Luo, N. Calabretta, G. Bernini, N. Ciulli, J. C. Sancho, S. Iordache, M. Farreras, Y. Becerra, C. Liou, I. Hussain, Y. Yin, L. Liu, and R. Proietti, "All-optical packet/circuit switching-based data center network for enhanced scalability, latency, and throughput," *IEEE Network*, vol. 27, pp. 14–22, November 2013.

[12] "Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model," standard, International Organization for Standardization, Geneva, CH, Nov. 1994.

[13] "IEEE standard for ethernet," *IEEE Std 802.3-2015 (Revision of IEEE Std 802.3-2012)*, pp. 1–4017, March 2016.

[14] ITU-T, "Network node interface for the synchronous digital hierarchy (SDH)," Recommendation G.707/Y.1322, International Telecommunication Union, Geneva, Jan. 2007.

[15] ATIS, "Synchronous Optical Network (SONET) – Basic Description Including Multiplex Structure, Rates, and Formats," Standart ATIS-0900105, Apr. 2015.

[16] ITU-T, "Interfaces for the optical transport network," Recommendation G.709/Y.1331, International Telecommunication Union, Geneva, June 2016.

[17] ITU-T, "Spectral grids for WDM applications: DWDM frequency grid," Recommendation G.694.1, International Telecommunication Union, Geneva, Feb. 2012.

[18] T. A. Strasser and J. L. Wagener, "Wavelength-selective switches for roadm applications," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 16, pp. 1150–1157, Sep. 2010.

[19] W. Samoud, *Performance analysis of hybrid opto-electronic packet switch*. PhD thesis, 2016. PhD Thesis directed by Ware, Cédric et Lourdiane, Mounia.

[20] J. R. de Almeida Amazonas, G. Santos-Boada, and J. Solé-Pareta, "Who shot optical packet switching?," in *Int. Conference on Transparent Optical Networks (ICTON)*, no. Th.B3.3, July 2017.

[21] Yang Chen, Chunming Qiao, and Xiang Yu, "Optical burst switching: a new area in optical networking research," *IEEE Network*, vol. 18, pp. 16–23, May 2004.

[22] Calient, INC, "S Series Optical Circuit Switch." Online: https://www.calient.net/products/s-series-photonic-switch/. Accessed: 2019-08-07.

[23] K. J. Barker, A. Benner, R. Hoare, A. Hoisie, A. K. Jones, D. K. Kerbyson, D. Li, R. Melhem, R. Rajamony, E. Schenfeld, S. Shao, C. Stunkel, and P. Walker, "On the feasibility of optical circuit switching for high performance computing systems," in *SC '05:*

Proceedings of the 2005 ACM/IEEE Conference on Supercomputing, pp. 16–16, Nov 2005.

[24] G. N. Rouskas and L. Xu, *Optical Packet Switching*, pp. 111–127. Boston, MA: Springer US, 2005.

[25] P. Delesques, T. Bonald, G. Froc, P. Ciblat, and C. Ware, "Enhancement of an optical burst switch with shared electronic buffers," in *2013 17th International Conference on Optical Networking Design and Modeling (ONDM)*, pp. 137–142, April 2013.

[26] A. Kimsas, H. Øverby, S. Bjornstad, and V. L. Tuft, "A cross layer study of packet loss in all-optical networks," in *Proceedings of AICT/ICIW*, 2006.

[27] K. Christodoulopoulos, D. Lugones, K. Katrinis, M. Ruffini, and D. O'Mahony, "Performance evaluation of a hybrid optical/electrical interconnect," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 7, pp. 193–204, March 2015.

[28] F. Yan, X. Xue, and N. Calabretta, "Hifost: a scalable and low-latency hybrid data center network architecture based on flow-controlled fast optical switches," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, pp. 1–14, July 2018.

[29] T. Segawa, S. Ibrahim, T. Nakahara, Y. Muranaka, and R. Takahashi, "Low-power optical packet switching for 100-Gb/s burst optical packets with a label processor and 8 x 8 optical switch," *J. Lightw. Technol.*, vol. 34, pp. 1844–1850, April 2016.

[30] W. Samoud, C. Ware, and M. Lourdiane, "Performance analysis of a hybrid optical-electronic packet switch supporting different service classes," *IEEE J. Opt. Commun. Netw.*, vol. 7, pp. 952–959, Sept 2015.

[31] P. J. Argibay-Losada, G. Sahin, K. Nozhnina, and C. Qiao, "Transport-layer control to increase throughput in bufferless optical packet-switching networks," *IEEE J. Opt. Commun. Netw.*, vol. 8, pp. 947–961, Dec. 2016.

[32] Cisco, INC, "Cisco Global Cloud Index 2014–2019." Online: https://bit.ly/2Kj8FvU, 2015. Accessed: 2019-08-07.

[33] Cisco, INC, "Cisco Global Cloud Index: Forecast and Methodology, 2016–2021 White Paper." Online: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html, 2018. Accessed: 2019-08-07.

[34] T. Wang, Z. Su, Y. Xia, and M. Hamdi, "Rethinking the data center networking: Architecture, network protocols, and resource sharing," *IEEE Access*, vol. 2, pp. 1481–1496, 2014.

[35] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication*, SIGCOMM '08, (New York, NY, USA), pp. 63–74, ACM, 2008.

[36] A. Andreyev, "Introducing data center fabric, the next-generation facebook data center network." Online: https://code.fb.com/production-engineering/introducing-data-center-fabric-the-next-generation-facebook-data-center-network/, Nov 2014. Accessed: 2018-07-17.

[37] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, "Data center tcp (dctcp)," in *Proceedings of the ACM SIGCOMM 2010 Conference*, SIGCOMM '10, (New York, NY, USA), pp. 63–74, ACM, 2010.

[38] S. Bensley, D. Thaler, P. Balasubramanian, L. Eggert, and G. Judd, "Data Center TCP (DCTCP): TCP Congestion Control for Data Centers," RFC 8257, RFC Editor, October 2017.

[39] N. Terzenidis, M. Moralis-Pegios, G. Mourgias-Alexandris, T. Alexoudi, K. Vyrsokinos, and N. Pleros, "High-port and low-latency optical switches for disaggregated data centers: The hipo[lambda]aos switch architecture [invited]," *J. Opt. Commun. Netw.*, vol. 10, pp. B102–B116, Jul 2018.

[40] N. Farrington, G. Porter, S. Radhakrishnan, H. H. Bazzaz, V. Subramanya, Y. Fainman, G. Papen, and A. Vahdat, "Helios: A hybrid electrical/optical switch architecture for modular data centers," in *Proceedings of the ACM SIGCOMM 2010 Conference*, SIGCOMM '10, (New York, NY, USA), pp. 339–350, ACM, 2010.

[41] A. Singla, A. Singh, K. Ramachandran, L. Xu, and Y. Zhang, "Proteus: A topology malleable data center network," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, Hotnets-IX, (New York, NY, USA), pp. 8:1–8:6, ACM, 2010.

[42] N. Benzaoui, J. M. Estarán, E. Dutisseuil, H. Mardoyan, G. D. Valicourt, A. Dupas, Q. P. Van, D. Verchere, B. Uščumlić, M. S. Gonzalez, P. Dong, Y. . Chen, S. Bigo, and Y. Pointurier, "CBOSS: bringing traffic engineering inside data center networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, pp. 117–125, July 2018.

[43] A. Minakhmetov, C. Ware, and L. Iannone, "TCP congestion control in datacenter optical packet networks on hybrid switches," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, pp. 71–81, July 2018.

[44] A. Minakhmetov, H. Chouman, L. Iannone, M. Lourdiane, and C. Ware, "Network-level strategies for best use of optical functionalities [invited]," in *Proceedings of IEEE 2018 20th International Conference on Transparent Optical Networks (ICTON)*, pp. 1–4, July 2018.

[45] A. Minakhmetov, C. Ware, and L. Iannone, "Optical Networks Throughput Enhancement via TCP Stop-and-Wait on Hybrid Switches," in *Proceedings of IEEE/OSA 2018 Optical Fiber Communications Conference and Exposition (OFC)*, pp. 1–3, March 2018.

[46] A. Minakhmetov, C. Ware, and L. Iannone, "Data Center's Energy Savings for Data Transport via TCP on Hybrid Optoelectronic Switches," *IEEE Photonics Technology Letters*, vol. 31, pp. 631–634, April 2019.

[47] A. Minakhmetov, A. Nagarajan, L. Iannone, and C. Ware, "On the Latencies in a Hybrid Optical Packet Switching Network in Data Center," in *Proceedings of IEEE/OSA 2019 Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1–3, March 2019.

[48] A. Minakhmetov, C. Ware, and L. Iannone, "Hybrid and Optical Packet Switching Supporting Different Service Classes in Data Center Network," in *Proceedings of IFIP 23rd Conference on Optical Network Design and Modelling (ONDM)*, pp. 1–6, May 2019.

[49] D. J. Blumenthal, P. R. Prucnal, and J. R. Sauer, "Photonic packet switches: architectures and experimental implementations," *Proc. IEEE*, vol. 82, pp. 1650–1667, Nov 1994.

[50] P. Gambini, M. Renaud, C. Guillemot, F. Callegati, I. Andonovic, B. Bostica, D. Chiaroni, G. Corazza, S. L. Danielsen, P. Gravey, P. B. Hansen, M. Henry, C. Janz, A. Kloch, R. Krahenbuhl, C. Raffaelli, M. Schilling, A. Talneau, and L. Zucchelli, "Transparent optical packet switching: network architecture and demonstrators in the keops project," *IEEE J. Sel. Areas Commun.*, vol. 16, pp. 1245–1259, Sep 1998.

[51] C. Ware, W. Samoud, P. Gravey, and M. Lourdiane, "Recent advances in optical and hybrid packet switching," in *Int. Conference on Transparent Optical Networks (ICTON)*, no. Tu.D3.4, (Trento, Italia), July 2016.

[52] S. Ibrahim and R. Takahashi, "Hybrid optoelectronic router for future optical packet-switched networks," in *Optoelectronics - Advanced Device Structures* (S. L. Pyshkin and J. Ballato, eds.), ch. 04, Rijeka: InTech, 2017.

[53] W. Samoud, C. Ware, and M. Lourdiane, "Investigation of a hybrid optical-electronic switch supporting different service classes," in *Photonics North*, vol. 9288, (Montreal, Canada), pp. 928809,1–6, May 2014.

[54] M. Weldon, "The future of dynamic deterministic networking." Online: https://www.ofcconference.org/en-us/home/news-and-press/ofc-video-library/?videoId=5754297359001, Mar. OFC 2018 Plenary, Accessed: 2018-07-17.

[55] Y. Yin, R. Proietti, X. Ye, C. J. Nitta, V. Akella, and S. J. B. Yoo, "LIONS: An awgr-based low-latency optical switch for high-performance computing and data centers," *IEEE J. Sel. Topics Quantum Electron.*, vol. 19, pp. 3600409–3600409, March 2013.

[56] W. Miao, F. Yan, and N. Calabretta, "Towards petabit/s all-optical flat data center networks based on WDM optical cross-connect switches with flow control," *J. Lightw. Technol.*, vol. 34, pp. 4066–4075, Sept. 2016.

[57] F. Yan, G. Guelbenzu, and N. Calabretta, "A novel scalable and low latency hybrid data center network architecture based on flow controlled fast optical switches," in *OFC*, p. W2A.23, Mar. 2018.

[58] X. Yu, H. Gu, K. Wang, M. Xu, and Y. Guo, "MPNACK: an optical switching scheme enabling the buffer-less reliable transmission," vol. 10244, 2017.

[59] P. J. Argibay-Losada, K. Nozhnina, G. Sahin, and C. Qiao, "Using stop-and-wait to improve tcp throughput in fast optical switching (fos) networks over short physical distances," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, pp. 1312–1320, April 2014.

[60] J. Wang, C. McArdle, and L. P. Barry, "Retransmission schemes for lossless transparent optical packet switching in large-scale datacentre networks," in *2016 IEEE FiCloud*, pp. 207–212, Aug. 2016.

[61] X. Ye, P. Mejia, Y. Yin, R. Proietti, S. J. B. Yoo, and V. Akella, "DOS - a scalable optical switch for datacenters," in *2010 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, pp. 1–12, Oct 2010.

[62] R. Takahashi, T. Nakahara, Y. Suzaki, T. Segawa, H. Ishikawa, and S. Ibrahim, "Recent progress on the hybrid optoelectronic router," in *2012 International Conference on Photonics in Switching (PS)*, pp. 1–3, Sept 2012.

[63] N. Terzenidis, M. Moralis-Pegios, G. Mourgias-Alexandris, K. Vyrsokinos, and N. Pleros, "High-port low-latency optical switch architecture with optical feed-forward buffering for 256-node disaggregated data centers," *Opt. Express*, vol. 26, pp. 8756–8766, Apr 2018.

[64] J. Wang, C. McArdle, and L. P. Barry, "Modelling and dimensioning of a high-radix datacentre optical packet switch with recirculating optical buffers," *Optical Switching and Networking*, vol. 23, pp. 67–81, 2017.

[65] T. Chu, L. Qiao, W. Tang, D. Guo, and W. Wu, "Fast, high-radix silicon photonic switches," in *Optical Fiber Communication Conference*, p. Th1J.4, Optical Society of America, 2018.

[66] A. Shacham, B. A. Small, O. Liboiron-Ladouceur, and K. Bergman, "A fully implemented 12x12 data vortex optical packet switching interconnection network," *J. Lightwave Technol.*, vol. 23, p. 3066, Oct 2005.

[67] Q. Cheng, A. Wonfor, J. L. Wei, R. V. Penty, and I. H. White, "Low-energy, high-performance lossless 8 x 8 SOA switch," in *2015 Optical Fiber Communications Conference and Exhibition (OFC)*, pp. 1–3, March 2015.

[68] A. Rylyakov, J. Proesel, S. Rylov, B. Lee, J. Bulzacchelli, A. Ardey, C. Schow, and M. Meghelli, "A 25 gb/s burst-mode receiver for low latency photonic switch networks," in *Optical Fiber Communication Conference*, p. W3D.2, Optical Society of America, 2015.

[69] V. Paxson, M. Allman, J. Chu, and M. Sargent, "Computing TCP's retransmission timer," RFC 6298, RFC Editor, June 2011.

[70] N. Agrawal, W. Bolosky, J. Douceur, and J. Lorch, "A five-year study of file-system metadata," *ACM Trans. Storage*, vol. 3, no. 3, 2007.

[71] A. Andreyev, "Introduction to facebook's data center fabric." Online: https://youtu.be/mLEawo6OzFM?t=175, Nov 2014. Accessed: 2018-07-17.

[72] E. Blanton, M. Allman, L. Wang, I. Jarvinen, M. Kojo, and Y. Nishida, "A conservative loss recovery algorithm based on selective acknowledgment (SACK) for TCP," RFC 6675, RFC Editor, August 2012.

[73] M. Allman, V. Paxson, and E. Blanton, "TCP congestion control," RFC 5681, RFC Editor, September 2009.

[74] J. Baliga, R. W. A. Ayre, K. Hinton, and R. S. Tucker, "Green cloud computing: Balancing energy in processing, storage, and transport," *Proceedings of the IEEE*, vol. 99, pp. 149–167, Jan 2011.

[75] N. Binkert, A. Davis, N. P. Jouppi, M. McLaren, N. Muralimanohar, R. Schreiber, and J. H. Ahn, "The role of optics in future high radix switch design," in *2011 38th Annual International Symposium on Computer Architecture (ISCA)*, pp. 437–447, June 2011.

[76] K. Lee, B. Sedighi, R. S. Tucker, H. Chow, and P. Vetter, "Energy efficiency of optical transceivers in fiber access networks [invited]," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 4, pp. A59–A68, Sept 2012.

[77] D. Larrabeiti, P. Reviriego, J. Hernández, J. Maestro, and M. Urueña, "Towards an energy efficient 10 Gb/s optical ethernet: Performance analysis and viability," *Optical Switching and Networking*, vol. 8, no. 3, pp. 131 – 138, 2011. Special Issue on Green Communications and Networking.

[78] J. Dean and L. A. Barroso, "The tail at scale," *Communications of the ACM*, vol. 56, pp. 74–80, 2013.

[79] G. Judd, "Attaining the promise and avoiding the pitfalls of TCP in the datacenter," in *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*, (Oakland, CA), pp. 145–157, USENIX Association, May 2015.

[80] N. Dukkipati and N. McKeown, "Why flow-completion time is the right metric for congestion control," *SIGCOMM Comput. Commun. Rev.*, vol. 36, pp. 59–62, Jan. 2006.

[81] K. Ramakrishnan, S. Floyd, and D. Black, "The addition of explicit congestion notification (ecn) to ip," RFC 3168, RFC Editor, September 2001. http://www.rfc-editor.org/rfc/rfc3168.txt.

[82] T. Henderson, S. Floyd, A. Gurtov, and Y. Nishida, "The newreno modification to tcp's fast recovery algorithm," RFC 6582, RFC Editor, April 2012. http://www.rfc-editor.org/rfc/rfc6582.txt.

[83] J. Luo, J. Jin, and F. Shan, "Standardization of low-latency tcp with explicit congestion notification: A survey," *IEEE Internet Computing*, vol. 21, pp. 48–55, Jan 2017.

[84] M. Alizadeh, "6.888: Lecture 3 data center congestion control." Online: https://people.csail.mit.edu/alizadeh/courses/6.888/slides/lecture3.pptx. Advanced Topics in Networking, MIT, Accessed: 2019-08-30.

[85] M. Noormohammadpour and C. S. Raghavendra, "Datacenter traffic control: Understanding techniques and tradeoffs," *IEEE Communications Surveys Tutorials*, vol. 20, pp. 1492–1525, Secondquarter 2018.

[86] "100Gb/s Réseau Internet Adaptative (100GRIA) FUI9 project," tech. rep., Dec. 2012.

Titre : Commutation de paquets optique et hybride multicouches

Mots clés : Réseaux à commutation de paquets, commutation de paquets, réseaux transparents, commutateurs optiques, commutateurs hybrides, TCP, contrôle de congestion, CCA, centre de données, réseaux de centres de données

Résumé : Les réseaux de télécommunication transparent constituent une étape de développement des réseaux entièrement électroniques. Les technologies de réseau de données actuelles utilisent déjà activement les fibres optiques et les réseaux transparents dans les réseaux centraux, métropolitains et résidentiels. Toutefois, ces réseaux reposent toujours sur la commutation électronique de paquets (EPS) pour le routage des paquets, qui rend obligatoire pour chaque paquet d'avoir une conversion de signal optique à électronique à optique (OEO). D'autre part, la commutation optique de paquets (OPS), qui semblait remplacer le système EPS, promet depuis longtemps des améliorations en termes de performances et de consommation d'énergie en s'éloignant des conversions OEO; Cependant, l'absence de buffers optiques pratiques rendait OPS extrêmement vulnérable aux contentions, entraînant une réduction des performances et empêchant de tirer profit des gains de l'OPS.

L'objetif de cette recherche est d'étudier la performance des réseaux avec des commutateurs tout optiques et hybrides, tandis que les activités de transmission côté serveur sont régies par des protocoles de contrôle de transport basés sur des algorithmes de contrôle de congestion (TCP CCA). Nous considérons que l'opération OPS pourrait être activée en utilisant un commutateur hybride, c.à.d. une solution au niveau de l'appareil, ainsi que des TCP CCA spécialement conçus, c.à.d. une solution au niveau du réseau, donnant naissance à des réseaux hybrides à commutation de paquets optique (HOPS).

Nous étudions les réseaux de centres de données (DCN) de type OPS, HOPS et EPS associés à différentes TCP CCAs en suivant les trois axes de la performance: débit, consommation d'énergie et latence. En ce qui concerne les TCP CCA, nous considérons non seulement les solutions existantes, mais également celles développées. Si Stop-And-Wait (SAW), Selective Acknowledgment (SACK), SACK modifié (mSACK) et Data Center TCP (DCTCP) sont déjà connus, Stop-And-Wait-Longer (SAWL) est présenté ici et conçu pour tirer le meilleur du HOPS DCN. Il est démontré que les solutions de commutateurs hybrides surpassent de manière significative les commutateurs tout optiques sans buffer et atteignent le niveau de commutateurs tout électroniques en termes de débit du reseau. En termes de consommation d'énergie, les solutions hybrides peuvent économiser jusqu'à 4 fois plus d'énergie de la commutation par rapport aux solutions tout électroniques. De plus, les DCN HOPS peuvent atteindre des latences moyennes à l'échelle des microsecondes, dépassant ainsi les EPS et se situant au même niveau que les OPS.

La question de l'introduction de classes de service dans HOPS DCN est examinée: on constate que les règles de commutation spécifiques en commutation hybride peuvent améliorer la performance de certaines classes sans pertes significatives d'autres.

Title : Cross-layer Hybrid and Optical Packet Switching

Keywords : Packet-switched networks, Packet Switching, Transparent networks, Optical Switches, Hybrid Switches, TCP, Congestion control, CCA, Data Center, Data Center Networks

Abstract : Transparent optical telecommunication networks constitute a development step from all-electronic networks. Current data network technologies already actively employ optical fibers and transparent networks in the core, metro, and residential area networks. However, these networks still rely on Electronic Packet Switching (EPS) for packets routing, constituting obligatory for each packet optical-to-electronic-to-optical (OEO) signal conversion. On the other hand, Optical Packet Switching (OPS), seemed to be as replacement of EPS, has long promised performance and energy consumption improvements by going away from OEO conversions; however, the absence of practical optical buffers made OPS highly vulnerable to contention, incurring performance reduction, and getting in the way of profiting from OPS gains.

The subject of this research lies in the investigation of the performance of OPS networks under all-optical and hybrid switches, while server-side transmission activities are regulated by Transport Control Protocols based on Congestion Control Algorithms (TCP CCAs). We consider that OPS could be enabled by use hybrid switch, i.e. device-level solution, as well by use of specially designed TCP CCAs, i.e. network-level solution, giving birth to Hybrid Optical Packet Switching (HOPS) networks.

We extensively study OPS, HOPS and EPS types of Data Center Networks (DCN) coupled with different TCP CCAs use by following the next three axes of DCN performance: Throughput, Energy Consumption, and Latency. As for TCP CCAs we consider not only existing but also newly developed solutions. If Stop-And-Wait (SAW), Selective Acknowledgment (SACK), modified SACK (mSACK) and Data Center TCP (DCTCP) are already known to the world, Stop-And-Wait-Longer (SAWL) is newly presented and is designed to bring the best out of the HOPS DCN. As a result, it is shown that hybrid switch solutions significantly outperform bufferless all-optical switches and reach the level of all-electronic switches in DCNs in terms of throughput. In terms of energy consumption, hybrid solutions can save up to 4 times on energy on switching compared to all-electronic solutions. As well HOPS DCNs can exhibit microseconds-scale average latencies, surpassing EPS and performing on the level with OPS.

The question of the introduction of Classes of Service to HOPS DCN is also investigated: it was found that class-specific switching rules to hybrid switch can ameliorate the performance of certain classes without almost performance loss in others.

Institut Polytechnique de Paris
91120 Palaiseau, France