Optimization of routing and wireless resource allocation in hybrid data center networks

Boutheina Dab

Doctor of Philosophy
University Paris-Est

Specialization

COMPUTER SCIENCE
(École Doctorale Mathématiques et Sciences et Technologies de l'Information et de la Communication (MSTIC))

presented by

Ms Boutheina Dab

Submitted for the degree of

Doctor of Philosophy of University Paris-Est

Title:

Optimization of Routing and Wireless Resource Allocation in Hybrid Data Center Networks

Defense: 5th July 2017

Committee:

Raouf Boutaba	**Reviewer**	**Full Professor, University of Waterloo - Canada**
Marcelo Dias de Amorim	**Reviewer**	**Research director, CNRS-LIP6-UMPC Sorbonne Universités, Paris - France**
Bernard Cousin	**President**	**Full Professor, IRISA, University of Rennes 1 - France**
Paul Muhlethaler	**Examiner**	**Research director, INRIA-Paris, EVA team - France**
Djamal Zeglache	**Examiner**	**Full Professor, Telecom SudParis, Evry - France**
Jérémie LEGUAY	**Examiner**	**Research engineer, Huawei Technologies - Paris - France**
Ilhem Fajjari	**Advisor**	**Research engineer, Orange Labs, Châtillon - France**
Nadjib Aitsaadi	**Supervisor**	**Full Professor, University Paris-Est, LIGM-CNRS, ESIEE Paris, Noisy-le-Grand - France**

UNIVERSITÉ PARIS-EST

THESE DE DOCTORAT DE
L'UNIVERSITE PARIS-EST

Spécialité

INFORMATIQUE
(École Doctorale Mathématiques et Sciences et Technologies de l'Information et de la Communication (MSTIC))

Présentée par

Mme. Boutheina Dab

Pour obtenir le grade de

DOCTEUR de l'UNIVERSITE PARIS-EST

Sujet de la thèse :

Optimisation du routage et de l'allocation de ressources sans fil dans les réseaux des centres de données Hybrides

soutenue le : 5 Juillet 2017

Jury :

Raouf Boutaba	**Rapporteur**	**Professeur, University of Waterloo - Canada**
Marcelo Dias de Amorim	**Rapporteur**	**Directeur de recherche, CNRS-LIP6-UMPC Sorbonne Universités, Paris - France**
Bernard Cousin	**Président**	**Professeur, IRISA, Université de Rennes 1 - France**
Paul Muhlethaler	**Examinateur**	**Directeur de recherche, INRIA-Paris - France**
Djamal Zeglache	**Examinateur**	**Professeur, Telecom SudParis, Evry - France**
Jérémie LEGUAY	**Examinateur**	**Ingénieur de recherche, Huawei Technologies - Paris - France**
Ilhem Fajjari	**Co-encadrante**	**Ingénieure de recherche, Orange Labs, Châtillon - France**
Nadjib Aitsaadi	**Directeur de thèse**	**Professeur, University Paris-Est, LIGM-CNRS, ESIEE Paris, Noisy-le-Grand - France**

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor, Prof. Nadjib Aitsaadi, for his continuous support, excellent guidance, precious recommendations. Your perpetual passion with research and enthusiasm had motivated me so much in hard moments. It was such a great experience working with you!

Special thanks go also for my co-advisor, Dr. Ilhem Fajjari, for her support, her precious assistance, and constructive advices. Your insightful discussions and instructions have importantly enriched this work. It has been always valuable for me to collaborate with you.

I would like to thank Prof. Raouf Boutaba and Dr. Marcelo Dias de Amorim for reviewing my thesis, and for the valuable feedback and constructive suggestions. I am also grateful to all the jury members, Prof. Bernard Cousin, Dr. Paul Muhlethaler, Prof. Djamal Zeglache and Dr. Jérémie LEGUAY, for their interest to examine my research work.

My special thanks go also for my friends and colleagues in the Lab, for the pleasant work atmosphere. Particularly, I would like to thank Zakia Khalfallah, Dalal Toudji, Farida Benaouda, Sarra Naffakhi, Rania Boussandel, Roua Touihri, Ichrak Amdouni, Fatima, Fetia Bannour, Sajid Mushtaq, Yazid Lyazidi, Karim Senouci, Ferhat, Safwan Alwan and Nicolas. I would also like to thank the staff of LiSSi, mainly Coumar Soupramanien and Katia Lambert for their help.

I am truly thankful for my nice friends, from outside the lab, for their continuous support, especially: Ahlam Bencheikh, Imen Boudhiba, Hiba Hajri, Amira Bezzina, Olfa Laabidi, Tasnim Hamza, Yosra Ghraibia, Fadwa Rekik, Hela Guesmi, Salma Matoussi, Mouna Ichawia and Zoubeyda. Thank you for all the joyful moments I have shared with you, and the positive energy you provide to me. I am so blessed for having you by my side during this journey.

I would like especially to convey my infinite gratitude to my dear parents, who always believed in me. Thank you for your unconditional love and constant support. All that I am, or hope to be, I owe to you. I am particularly indebted to my lovely sisters Marwa, Safa and Fatma, and to my amazing brother Bilel. Thanks for standing by my side during difficult times and for the wonderful moments we always share together. A special thanks go for my lovely grand mothers, for their sincere prayers and love, to my dear aunts and uncles, in Tunisia, my cousins Faten, Nesrine, Sana, Wafa, Safa, Meryem, Hanen, Oussama and Daly :-) I am so grateful for having you by my side.

To my parents Mohamed and Salima
To the soul of my grandfather
I dedicate this work

Abstract

The high proliferation of smart devices and online services allows billions of users to connect with network while deploying a vast range of applications. Particularly, with the advent of the future 5G technology, it is expected that a tremendous mobile and data traffic will be crossing Internet network. In this regard, Cloud service providers are urged to rethink their data center architectures in order to cope with this unprecedented traffic explosion. Unfortunately, the conventional wired infrastructures struggle to resist to such a traffic growth and become prone to serious congestion problems. Therefore, new innovative techniques are required.

In this thesis, we investigate a recent promising approach that augments the wired Data Center Network (DCN) with wireless communications. Indeed, motivated by the feasibility of the new emerging 60 GHz technology, offering an impressive data rate ($\approx$ 7 Gbps), we envision, a Hybrid (wireless/wired) DCN (HDCN) architecture. Our HDCN is based on i) Cisco's Massively Scalable Data Center (MSDC) model and ii) IEEE 802.11ad standard. Servers in the HDCN are regrouped into racks, where each rack is equipped with a: i) Ethernet top-of-rack (ToR) switch and ii) set of wireless antennas. Our research aims to optimize the routing and the allocation of wireless resources for inter-rack communications in HDCN while enhancing network performance and minimizing congestion. The problem of routing and resource allocation in HDCN is NP-hard. To deal with this difficulty, we will tackle the problem into three stages. In the first stage, we consider only one-hop inter-rack communications in HDCN, where all communicating racks are in the same transmission range. We will propound a new wireless channel allocation approach in HDCN to harness both wireless and wired interfaces for incoming flows while enhancing network throughput. In the second stage, we deal with the multi-hop communications in HDCN where communicating racks can not communicate in one single-hop wireless path. We propose a new approach to jointly route and allocate channels for each single communication flow, in an online way. Finally, in the third stage, we address the batched arrival of inter-rack communications to the HDCN so as to further optimize the usage of wireless and wired resources. For that end, we propose: i) a heuristic-based and ii) an approximate, solutions, to solve the joint batch routing and channel assignment. Based on extensive simulations conducted in QualNet simulator while considering the full protocol stack, the obtained results for both real workload and uniform traces, show that our proposals outperform the prominent related strategies.

Key Words

Cloud Computing, Hybrid Data Center Networks, wireless communications, 60 GHz technique, IEEE 802.11ad standard, routing, resource allocation, optimization.

Résumé

Avec l'arrivée de la prochaine technologie 5G, des billions de terminaux mobiles seront connectés et une explosion du trafic de données est ainsi prévue. A cet égard, les fournisseurs des services Cloud nécessitent les infrastructures physiques efficaces capables de supporter cette croissance massive en trafic. Malheureusement, les architectures filaires conventionnelles des centres de données deviennent staturées et la congestion des équipements d'interconnexion est souvent atteinte. Dans cette thèse, nous explorons une approche récente qui consiste à augmenter le réseau filaire du centre de données avec l'infrastructure sans fil. En effet, nous exploitons la nouvelle technologie 60 GHz, qui assure un débit important de l'ordre de 7 Gbits/s afin d'améliorer la QoS. Nous concevons une architecture hybride (filaire/sans fil) du réseau de centre de données basée sur: i) le modèle "Cisco's Massively Scalable Data Center" (MSDC), et ii) le standard IEEE 802.11ad. Dans une telle architecture, les serveurs sont regroupés dans des racks, et sont interconnectés à travers un switch Ethernet, appelé top-of-rack (ToR) switch. Chaque ToR switch possède plusieurs antennes utilisées en parallèle sur différents canaux sans fil. L'objectif final consiste à minimiser la congestion du réseau filaire, en acheminant le maximum du trafic sur les liens sans fil. Pour ce faire, cette thèse se focalise sur l'optimisation du routage et de l'allocation des canaux sans fil pour les communications entre les racks, au sein d'un centre de données hybride (HDCN). Ce problème étant NP-difficile, nous allons procéder en trois étapes. En premier lieu, on considère le cas des communications à saut unique, où les racks sont placés dans le même rayon de transmission. Nous proposons un nouvel algorithme d'allocation des canaux sans fil dans les HDCN, qui permet d'acheminer le maximum des communications en sans fil, tout en améliorant les performances réseau en termes de débit et délai. En second lieu, nous nous adressons aux communications multi-sauts, où les racks ne sont pas dans le même rayon de transmission. Nous allons proposer une nouvelle approche optimale traitant conjointement le problème du routage et de l'allocation de canaux sans fils dans le HDCN, en mode en ligne. En troisième étape, nous proposons un nouvel alogorithme qui calcule conjointement le routage et l'allocation des canaux pour un ensemble des communications arrivant en bloc (i.e., mode batch). En utilisant le simulateur QualNet, les résultats obtenus montrent que nos propositions améliorent les performances réseau.

Mots-clés :

Cloud Computing, centres de données hybrides, communications sans fil, technique 60 GHz, standard IEEE 802.11ad , routage, allocation de resources, optimisation

Table of contents

Chapter 1

Introduction

Contents

Thanks to the advent of the long-awaited fifth generation (5G) mobile networks, mobile data and online services are becoming widely accessible. Discussions of this new standard have taken place in both industry and academia to design this emerging architecture. The main objective is to ensure, by 2020 [1], the capability to respond to the different applications needs such as videos, games, web searching, etc, while ensuring a higher data rate and an enhanced Quality of Service (QoS). Whilst no official standardization is yet delivered for 5G, experts assure that, the impressive proliferation of smart devices will lead to the explosion of traffic demand. Billions of connected users are expected to deploy a myriad of applications.

In this respect, recent statistics elaborated by CISCO Visual Networking Index (VNI) [2] highlight that the annual global IP traffic will roughly triple over the next 5 years, and will reach 2.3 zettabytes by 2020. More specifically, it is expected that smart phones traffic will impressively increase from 8% in 2015 to 30% of the total of IP traffic in 2020. As it is depicted through Figure. 1.1, mobile data traffic per month will grow from 7 Exabytes in 2016 to 49 Exabytes by 2021. In particular, tremendous video traffic will be crossing IP networks to reach 82% of the totality of IP traffic. It is also expected that the number of connected mobile devices will be more than

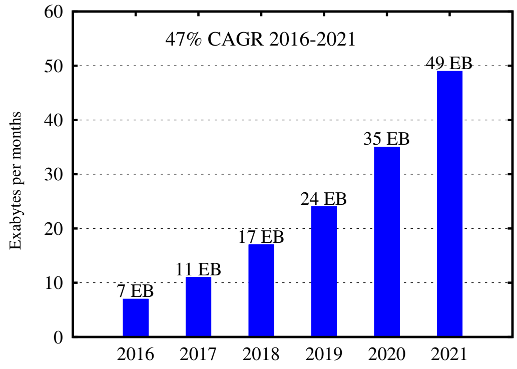


Figure 1.1: Mobile traffic growth

three times the size of the global population by 2020. In this regard, future networks are anticipated to support and connect plenty of devices, while offering higher data rate and lower latency.

To cope with this unprecedented traffic explosion, the service providers are urged to rethink their network architectures. In fact, efficient scalable physical infrastructures, e.g., data centers (DCs), are required to support the drastically increasing number of both online services and users.

To manage their DCs infrastructure, many of giant service tenants are resorting to virtualization technologies making use of Software Defined Networking (SDN) and Network Functions Virtualization (NFV) [3]. On one hand, SDN controllers offer the opportunity to implement more powerful algorithms thanks to a real-time centralized control leveraging an accurate view of the network. Indeed, thanks to the separation of the forwarding and the control planes, the managements complexity of the network infrastructure is considerably reduced while providing tremendous computational power compared to legacy devices. On the other hand, thank to NFV paradigm, network functions and communication services are first softwarized and then cloudified, so that they can be on demand orchestrated and managed as cloud-native IT applications. It is straightforward to see that these approaches are complimentary. They offer a new way to design and manage data centers while guaranteeing a high level of flexibility and scalability.

The new emerging SDN and NFV technologies requires scalable infrastructures. To that end, a great deal of efforts have been devoted to the design of efficient DC architectures. Indeed, Internet giants ramped up their investment in data centers/IT infrastructures and poured in billions of dollars to widen their global presence and improve their competitiveness in the Cloud market.

In this context, the latest Capital Expenditure (CAPEX) of the five largest-scale Internet operators, Apple, Google, Microsoft, Amazon and Facebook, increased by 9.7% in 2016 in order to invest in designing their DCs [4]. Over the past years, these companies have spent, in total, a capital of $\$115$ billions, to build out their DCs. For instance, Google has invested millions of dollars in expanding its data centers spread all over the world: Taiwan, Latin America, Singapore, etc. Facebook has started, since 2010, building out its own DCs in Altoona, Iowa and North Carolina.

In this regard, efficiently designing data centers is a crucial task to ensure scalability required to meet today's massive workload of Cloud applications. Moreover, it is mandatory to deploy the proper mechanisms for routing and resource allocation to communication flows in DCs.

To deal with these challenges, we investigate, in this thesis, a radically new methodology changing the design of traditional Data Center Network (DCN) while ensuring scalability and enhancing performance. Then, we address the problem of routing and resource allocation in DCNs. To that end, we will propose new routing and resource allocation strategies so as to minimize congestion effects and enhance network performance in terms of throughput and end-to-end delay.
The rest of this chapter is organized as follows. First, we will introduce the data center networking concept and highlight the main challenges faced to conventional wired DCN. Secondly, we will present the recent DCN architecture solutions. Afterwards, we will describe the problem addressed by our current research work. Finally, we will summarize our contributions.

1.1 Data Center Designing

Over the last decade, Cloud computing has been rapidly emerging to deeply impact our way of life. It is a promising technology entailing a service model that enables tenants to acquire and/or release on demand resources according a specific Service-Level Agreement (SLA). This service mode, commonly known as `pay-to-use` model, determines the fashion in which enterprises deploy IT infrastructure. One of the most immediate benefits of using Cloud services is the ability to speedily increase infrastructure capacity while alleviating maintenance costs.

Nevertheless, Cloud computing requires a performant underlying network infrastructure that is able to efficiently carry the tremendous amount of traffic circulating over a large number of servers. In fact, it has been highlighted that the number of servers owned by some Cloud operators can exceed one million [5]. Therefore, designing such huge environments based on traditional network is not judicious, and may induce extra maintenance costs.

In this context, Data-Center-as-a-Service (DCaaS) reveals as a crucial Cloud service mode. Actually, a Cloud infrastructure is constituted by a set of data centers interconnected to each others. Accordingly, a DC is defined as the home hosting tens to hundreds of thousands of servers, where each one is characterized by its: i) CPU, ii) memory, iii) network interfaces, and iv) local high data rate [5]. Typically, servers are regrouped into racks, and the latter are packaged into clusters consisting of thousands of hosts that are connected with high-bandwidth links. Such a design guarantees high performances while supporting today's large-scale applications, such as social networks and computing tasks. The interconnection of the large number of hosted servers and switches with high-speed communication links, in a DC, is ensured based on the Data Center Network (DCN).

1.1.1 Data Center Network

Data Center Network (DCN) represents the infrastructure interconnecting the physical resources (i.e., servers, switches, etc.) within the same DC, using high speed communication links (i.e., cables, optical fibers), according to a specific topology. Basically, the DCN is defined by its: i) network topology, ii) routing/switching equipments and iii) network protocols. DCN plays a decisive role in computing and deeply impacts the efficiency and performance quality of the applications. Data center networking brings many benefits to Cloud providers. First, it enables the interconnection between numerous servers and arranges thousands of hosts in an efficient topology. Moreover, DCN can support virtualization technique, so that servers can host many virtual machines.

Conventionally, a data center network is based on a traditional multi-tier topology. It consists of a multi-rooted tree-like architecture, mainly formed by: i) servers and ii) three layers (i.e., core, aggregation and edge) of switches. Typically, traditional DCN interconnects servers while making use of electronic switching with a limited number of ports. Hereafter, we will present each hardware component of the multi-tier DCN architecture.

1. **`Servers`**: represent the core physical components of the DCN. They directly impact the network performance in the DC since they are responsible for massive data processing, storing and transmission.

2. **`Racks`**: are the container supporting servers, switches, and cables, in a way that saves space and simplifies resource management.

3. **`Switches`**: represent the backbone of the data center network. They are regrouped into three layers, in a top-down manner: i) core, ii) aggregation and iii) edge, switch layers. Core switches are used for inter-DCN connections, as their up-link ports are used to connect the DCN to the Internet. Aggregation switches connect distant servers belonging to different racks, and ensure, hence, inter-rack communications. The core and aggregation switches interconnect with 10 Gbps links while logically forming bipartite graphs. Finally, the servers in each rack are connected directly to an edge switch, placed in the top of the rack (i.e., ToR switch) with 1 Gbps links. Note that the performance of such an equipment strongly depends on the switching speed and the number of ingress/egress ports.

4. **`Cables`**: are the elements that interconnect all the components (i.e., switches, servers) with each others and that transport electricity or optical signals. Commonly, cabling in conventional wired DCN is based on Ethernet standard.

The traditional multi-layer DCN architecture is illustrated through Figure. 1.2.

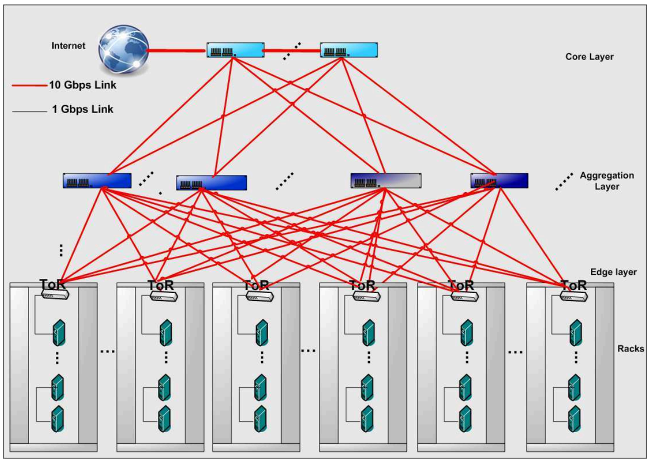


Figure 1.2: Conventional three-layer DCN architecture

1.1.2 Data Center Network Challenges

To meet the increasing demand of cloud services, huge traffic is susceptible to transit within DCNs. Moreover, thanks to virtualisation technique, multi-tenancy emerges as a promising way to share instances of computing resources among multiple tenants (i.e., group of users). Unfortunately, both the high availability of data and the elasticity of resource use induce important load over-subscription. DCN infrastructures are thus vulnerable to serious network congestion and resource contention problems. Actually, traditional DCN architecture is not well suited for Cloud data centers and cannot meet the increasing demand of online services.

In summary, traditional DCN architecture has several inherent drawbacks as follows.

- **Limited link capacity:** The available bandwidth in DCN is limited, which results in oversubscription. For instance, up to 40 servers can be encompassed into a single rack and connected to only one ToR switch with 1 Gbps links. The ToR is connected to an aggregation switch using 10 Gbps links. Therefore, links connecting ToRs to aggregation switches are highly oversubscribed with a ratio of 1:4 [6].

- **Unbalanced utilization:** Usually, servers, in traditional DCNs, are allocated for various applications in a static manner, according to the maximum requested traffic. In doing so, resource utilization is not balanced. Moreover, the Spanning Tree Protocol (STP) protocol is conventionally used to select a single short path regardless the potential over-subscription.

- **Scalability challenge:** The hierarchical topology of DCN is not able to cope with scalability challenge, since the unique way to scale such an architecture is to increase the number of network devices. However, this solution results in high construction costs.

- **Traffic un-predictability:** The un-predictability of traffic and the dynamic flow arrival raise greater challenges regarding resource managing. In fact, although the number of elephant flows remains, in general, relatively low, it, indeed, entails 50% of the total traffic in DCN [5].

- **Weak flexibility:** The maximum size of the DCN depends on the number of switch ports. Therefore, if no port is free, then some switches have to be replaced by others with more ports. Obviously, this alternative is time and cost consuming.
- **Cabling complexity:** The number of cables deployed in a DCN can be tremendous if the latter scales to a large size. Therefore, cabling task becomes very hard to fulfill as new servers are added, which is strongly challenging for DC providers.

To provide Cloud service with high quality, modern DCNs have to satisfy several criteria. First of all, data centers need to be easy to transport and deploy, in order to guarantee flexibility according to business requirements. Secondly, DCNs need to put an end to the the hard resource commitment by efficiently balancing the utilization of different servers and preventing them from being idle. More importantly, DCNs have to be, at the same time, scalable and efficient enough to handle the growing Cloud services and to cope with the increasing size of DCs.

These challenges have garnered both academic and industrial research attention. In fact, top international IEEE and ACM conferences on computer science such as SIGCOMM, MobiCom, INFOCOM [6] [7] [8], and leading international journals, like [9] [10], have already addressed the issues relevant to DCN architecture and started publishing DCN related papers. Furthermore, several institutions such as MIT, Stanford University, Google, Microsoft, Facebook and many others, have devoted specific research teams to focus on DCN architecture research work.

Hereafter, we will introduce the main adopted DCN solutions.

1.2 Data Center Network Solutions

During the last few years, a great deal of research efforts have been devoted to designing efficient DCN topologies, able to rapidly scale and cope with the tremendous unbalanced traffic load.

One **first** solution, consists in over-dimensioning the traditional data center network. For example, some recent research approaches such as VL2 architecture designed by Microsoft in [11] and the DCN propounded in [12], resort to combining many core links and switches while making use of multi-path routing in order to alleviate the congestion in the DCN core (i.e., switches). Nevertheless, even if this approach seems to be efficient in the short-term, it comes, actually, with implementation complexity and material cost due to the expensive investment and the heaviness of network management. In fact, link density in some of such designs [12] may make cabling task extremely challenging. Moreover, some strategies increase the wired link capacity to reach 40 Gbps so as to boost DCN performance. However, CISCO [13] has found out that using Multi-Gigabytes is expensive since the power consumption of 40 Gbps optics is more than 10X a single 10 Gbps.

Secondly, some other recent approaches introduced new advanced DCN infrastructures dealing with load concentration issue. For instance, new CLOS-based architectures [14], like FatTree [15] and VL2 [16], or BCube [17], have been propounded in hope to balance the load on the DCN using redundant multi-gigabytes wired links, and multi-port switches. However, despite the increased offered data rate, the wired DCN topologies are still facing challenges in term of flexibility and congestion issues. For example, two servers belonging to different racks need to pass through the upper-level links while communicating which each others, even if they are geographically close.

Third, to deal with scalability and congestion issues, a recent promising approach has investigated the possibility of augmenting the wired DCN with high-speed links in order to provide extra bandwidth and boost network performance. In the literature, DCN augmentation can mainly be achieved in two ways: i) using **optical** devices, or ii) using **wireless** antennas.

Optical DCN (O-DCN) is a DCN architecture that makes use of optical switches and cables in order to easily establish high-speed connections. O-DCN can be either fully optical [18] [19] or hybrid (i.e., optical/Ethernet) O-DCN [20]. Although they ensure on-demand flexible links with higher bandwidth compared to the traditional Ethernet links, O-DCNs require enough space above racks and height-restricted ceiling. The latter is not guaranteed in real DC environment. Moreover, they entail high manual cost and cabling complexity for large scale networks.

In this regard, wireless augmented DCN has been proposed to get rid of the aforementioned challenges. Basically, it relies, in most of cases, on wireless 60 GHz technique and places wireless antennas on top-of-racks for inter-rack communications. Similarly, such an augmented architecture figures out in two kinds: i) fully wireless DCN, and ii) hybrid DCN. A fully wireless DCN deploys only wireless devices and eliminates wired links [21]. The Hybrid DCN (HDCN), deploys on each ToR both wireless antennas and wired links. HDCN harness both wireless and wired interfaces to considerably improve the performance of DCN in terms of bandwidth and latency.

In this thesis, we resort to a hybrid (wireless/wired) DCN architecture. In doing so, traffic can be forwarded over wireless and/or wired links. Specifically, we make use of 60 GHz wireless technology to alleviate the congestion load. In fact, this technique, operating in the unlicensed band of $57 - 64$ GHz, is commonly deployed for HDCN and ensures a notable high data rate ($\approx$ 6.7 Gbps). Moreover, augmenting the wired DCN with a wireless infrastructure enhances the flexibility, as wireless links can be dynamically and easily established in on-demand manner.

Nevertheless, despite the aforementioned advantages of the hybrid DCN architecture, it is faced to several challenges. First, the number of wireless channels available in the physical layer and their bandwidth capacities are limited. Second, the 60 GHz technology guarantees high data rate signals only for a short range (≈ 10 meters) due to the strong attenuation. Thus, wireless channels scheduling is a challenging task in modern hybrid DCNs. Finally, the wireless links are prone to high interference and noise factors in a real DCN, which strongly impact the quality of signal for cloud services.

In this thesis, we will deal with the two first challenges by designing a hybrid DCN architecture based on: i) IEEE 802.11ad (wireless) and ii) Ethernet (wired) standards. To tackle the last challenge, we address the problem of routing and wireless channel allocation in HDCN while considering interference constraint. Our focus is to propose new efficient algorithms able to enhance DCN throughput. Our solutions should take into account the physical constraints of HDCN environment, such as interference, short transmission range, flexibility and scalability.

1.3 Problem statement

Motivated by the feasibility and the facility of 60 GHz technology deployment in DCNs [6], we envision, in this thesis, a HDCN architecture based on i) Cisco's Massively Scalable Data Center (MSDC) model [22] and ii) IEEE 802.11ad standard [23]. In our HDCN, each rack is equipped with i) One Top-of-Rack (ToR) switch interconnecting servers through wired links and ii) four 2D beamforming antennas (Transmission Units (TU)) supporting IEEE 802.11ad. Note that the use of the beamforming technique improves the coverage distance while mitigating interference effects. Moreover, each TU is configured with a dedicated channel and only 4 wireless channels are available in IEEE 802.11ad standard. Besides, our HDCN architecture guarantees the load balancing in the wired links by making use of the Equal Cost Multi-Path (ECMP) [22] protocol coupled with Open Shortest Path First (OSPF) protocol.

In this thesis, we tackle the problem of wireless and wired resource allocation in our hybrid MSDC architecture. Specifically, we focus on inter-rack communications in HDCN. The latter can occur either on one-hop link, when the communicating racks are placed close enough to each other, or through multi-hop links if they are not within the same transmission range. Consequently, efficient mechanisms are needed for: i) resource allocation for one-hop communications, and ii) joint routing and resource allocation for multi-hop communications, in HDCN. More specifically, our purpose is to harness both the wireless and wired interfaces to carry inter-rack communications, in such a way that enhances the DCN bandwidth by minimizing the end-to-end delay. In this regard, we put forward a Centralized Controller (CC), hosting the control plane, that monitors the traffic in the HDCN and computes: i) the optimized wireless channel allocation for one-hop flows and ii) the joint routing and channel assignment for each multi-hop communications. Our proposals have to take into consideration:

- Interference constraint: Prohibiting intra-flow interference and minimizing inter-flow one.
- Wireless resource limitations: Only four wireless antennas are available on each ToR switch using 4 orthogonal channels of IEEE 802.11ad standard.
- End-to-end transmission delay: Minimizing the transmission and re-transmission delay while allocating channels.

- Congestion level upon ToR switches: Alleviating congestion by balancing the load.

The aforementioned constraints endorse the hardness of the routing and resource allocation problems. Therefore, we deal with combinatorial optimization and integer linear programming formulations in order to obtain optimized solutions.

1.4 Thesis contributions

In this section, we will outline the main contributions of this thesis.

- **A survey of data center network architectures**
  We will provide an in-depth overview of the architectures of data center networks. Mainly, we will classify DCN architecture into: i) switch-centric DCN, ii) server-centric DCN, and iii) enhanced (optical and wireless) DCN. In the first group, we will present the main hierarchic wired data center network topologies found in the literature, while discussing their main features. In the second group, we will review the server-centric DCN structures and highlight the major advantages and drawbacks. In the third group, we will present the optical and wireless enhanced DCN architectures found in the literature. We will show both the benefits and challenges of these HDCN architectures. Afterwards, we will offer a comparison between the different designs of the taxonomy while presenting the future research direction. Finally, with regard to this comparison, we will present our hybrid (i.e., wireless/wired) data center network architecture that we conceive in this thesis. We will detail the network simulation results of: i) our implementation of IEEE 802.11ad standard, and ii) Beamforming technique deployment, to validate the feasibility of 60 GHz communications in HDCN.

- **A survey of routing and channel allocation approaches in HDCN**
  We will provide an in-depth overview on both one-hop and muti-hop intra-HDCN communication algorithms found in the literature. We can classify them into three main groups. The first group includes all the wireless channel allocation strategies dealing with inter-rack communications in one single hop. The second category comprises the algorithms dealing with joint routing and channel assignment problem for multi-hop communications in an online manner. Specifically, in these strategies, each single flow request is processed in sequential way as it arrives. The third group concerns the approaches addressing the joint routing and channel assignment problem in batch mode. In other words, these strategies process a set of communication flows simultaneously in order to deal with the batched arrivals of flows and to guarantee a more efficient use of the wireless and wired resources in the DCN. Finally, we conclude this chapter by providing a comparison between the different related work strategies and we will explain the main differences with respect to the problematic of this thesis.

- **Proposed routing and resource allocation strategies in HDCN**
  To address the routing and resource allocation problems detailed in section 1.3, we will propose a series of routing and resource allocation algorithms in HDCN. Particularly, we will propound a new algorithm in each group of the aforementioned taxonomy. In fact, due to the complexity of resource allocation for inter-rack communications in hybrid DCNs, we proceed, in this thesis, to dividing the entire problem into three stages. In the first stage, we will consider only one-hop inter-rack communications in HDCN by assuming that the communicating racks are placed in the same transmission range. We will propose for this case, a new wireless channel allocation approach in HDCN to harness both wireless and wired interfaces for incoming flows while enhancing network throughput. In the second stage, we deal with the multi-hop communications in HDCN where communicating racks can not communicate with one single wireless link. We will propound a new approach to jointly route and allocate channels for each communication flow in the HDCN, in an online way. Finally, in the third stage, we handle the batch arriving of multi-hop inter-rack communications in HDCN. We propose two algorithms to solve the joint batch routing and channel assignment.
  Hereafter, we will detail the problem studied in each stage and the corresponding solutions.

  1. In the first stage, we only focus on communication flows between racks in the same wireless transmission range. Our objective is to minimize the end-to-end delay in the HDCN. To do so, we consider interference constraint, prohibiting the assignment of one wireless channel to more than one wireless link in the interference area. To deal with this challenge, we propose a new algorithm, denoted by resource allocation algorithm based on **Graph Coloring in Hybrid Data Center Network** (`GC-HDCN`) [24], maximizing the total throughput supported in the DCN. The main idea behind `GC-HDCN` is to maximize the proportion of communication requests transiting over the wireless infrastructure and the rest will be transmitted over the wired infrastructure. In doing so, the end-to-end delay of communications and the congestion of wired infrastructure are minimized. The problem is formulated as minimum graph coloring which is **NP-Hard**. `GC-HDCN` makes use of i) column generation and ii) branch and price optimization schemes to resolve the assignment of wireless channels. Based on extensive simulations with QualNet simulator considering all the protocol stack layers, the obtained results outperform the related prominent strategies. Despite the efficiency of our proposed algorithm, a one-hop wireless link is not enough to support traffic. In fact, flows in real DCs are diverse and may occur between geographically distant racks.

  2. In the second stage, we deal with multi-hop inter-rack communications in HDCN. In order to overcome the short range limitations of 60 GHz technique, we tackle the challenge of jointly i) routing and ii) allocating wireless channels for inter-rack flows,

while considering beamforming antennas. We propound an advanced **Joint Routing and Channel Assignment algorithm for HDCN** (JRCA-HDCN) [25]. To do so, we, first, formulate the problem as a minimum weight perfect matching. Then, our resolution is based on Edmond's Blossom algorithm. JRCA-HDCN aims to maximize the throughput of intra-HDCN communications over the wireless and/or wired infrastructure. Mainly, JRCA-HDCN takes into consideration both the i) length of IP queues (waiting delay) in each relay node and ii) level of wireless interferences (retransmission delay). JRCA-HDCN is an online approach since it sequentially computes the best hybrid (wireless and/or wired) path for each on-demand flow between a source rack S to a destination rack D. Unfortunately, it is unable to handle the batched arrival of communication requests. In fact, workload traces of real data centers, such as Facebook's DC, show that many flow requests are likely to arrive at the same time to the network. Therefore, it is more judicious to simultaneously process all the arriving requests in the batch so that an efficient use of wireless and wired resource in the HDCN is guaranteed.

3. In the third stage, we deal with the **Joint Batch Routing and Channel** Assignment problem (JBRC) in HDCN, to handle the batched arrivals of flow requests. We formulate JBRC using an advanced Multi-Commodity Flow (MCF) model, where each commodity corresponds to a communication demand. The objective of JBRC is to find for each batch of flow requests, the corresponding hybrid (wireless and/or wired) routing paths. JBRC bears an optimization objective of minimizing the end-to-end delay over all the links of the hybrid routing paths. To solve JBRC, we propose three solutions. First, an exact approach BR-HDCN able to compute optimal hybrid paths for small instances of JBRC problem. Second, to solve large instances of JBRC in a reasonable time, we propose a heuristic-based solution JBH-HDCN able to reduce complexity. However, JBH-HDCN doesn't guarantee a near-to-optimal solution. Therefore, we propose, third, an approximate scalable approach SJB-HDCN that considers the dimension challenge and converges to a feasible solution with a guaranteed precision. The obtained results are very satisfactory.

1.5 Thesis outline

The remainder of this manuscript is organized as follows. In chapter 2, we will present a taxonomy of the different data center network architectures. Next, in chapter 3, we will discuss the different routing and resource allocation strategies in HDCN. Besides, chapter 4 will detail the wireless channel allocation approach in HDCN based on Graph Coloring GC-HDCN which deals with one-hop communications. Chapter 5 will present the Joint Routing and Channel Assignment in HDCN (JRCA-HDCN) approach to process multi-hop communications in online mode. Afterwards, we will detail, in chapter 6, the Joint Batch Routing and Channel allocation problem `JBRC` and detail our: i) exact, ii) heuristic and iii) approximate proposed solutions. Finally, chapter 7 will conclude this thesis and will give an insight on our ongoing and future work in the field.

Chapter 2

Architectures of Data Center Networks: Overview

Contents

2.1 Introduction

To deal with the widespread use of cloud services and the unprecedented traffic growth, the scale of the DC has importantly increased. Therefore, it is crucial to design novel efficient network architectures able to satisfy the requirements on bandwidth. As a key physical infrastructure, DCN designing has widely been a hot research focus.

This chapter reviews the main DCN architectures propounded in the literature. To do so, a taxonomy of DCN designs will be proposed, while analyzing in depth each structure of the given

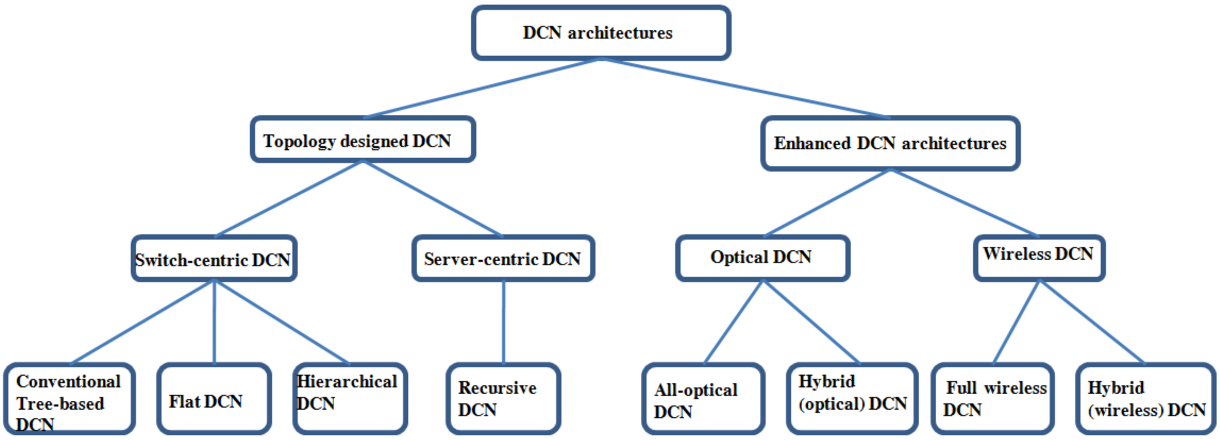


Figure 2.1: Taxonomy of DCN architectures

classification. Then, we will provide a qualitative comparison between these different DCN groups. Finally, we will present our DCN architecture considered in this thesis.

2.2 Taxonomy of data center network architectures

In this section, we present a taxonomy of the existent DCN architectures with a detailed review of each drawn class. In general, several criteria have to be considered to design robust DCNs, namely, high network performance, efficient resource utilization, full available bandwidth, high scalability, easy cabling, etc. To deal with the aforementioned challenges, a panoply of solutions have been designed. Mainly, we can distinguish two research directions. In the first one, wired DCN architectures have been upgraded to build advanced cost-effective topologies able to scale up data centers. The second approach has resorted to deploying new network techniques within the existing DCN so as to handle the challenges encountered in the prior architectures. Hereafter, we will give a detailed taxonomy of these techniques.

2.2.1 Classification of DCN architectures

With regard to the aforementioned research directions, we can identify three main groups of DCN architectures, namely, **switch-centric** DCN, **server-centric** DCN, and **enhanced** DCN. Each group includes a variety of categories that we will detail hereafter.

- **Switch-centric DCN architecture**: Switches are, mostly, responsible for network-related functions, whereas the servers handle processing tasks. The focus of such a design is to improve the topology so as to increase network scale, reduce oversubscription and speed up flow transmission. Switch-centric architectures can be classified into five main categories according to their structural properties:

1. Traditional tree-based DCN architecture: represents a specific kind of switch-centric architecture, where switches are linked in a multi-rooted form.
2. Hierarchic DCN architecture: is a switch-centric DCN where network components are arranged in multiple layers. Each layer characterizes traffic differently.
3. Flat DCN architecture: compresses the three switch layers into only one or two switch layers, in order to simplify the management and maintenance of the DCN.

- **Server-centric DCN architecture**: Servers are enhanced to handle networking functions, whereas switches are used only to forward packets. Basically, servers are simultaneously end-hosts and relaying nodes for multi-hop communications. Usually, server-centric DCN are recursively defined multi-level topologies.

- **Enhanced DCN architecture**: Is a specific DCN which is tailored for future Cloud computing services. Indeed, the future research direction attempts to deploy networking techniques so as to deal with wired DCN designs limitations. Recently, a variety of technologies have been used in this context, namely, optical switching, and wireless communications. Accordingly, we distinguish two main classes of enhanced DCN architectures:

  1. Optical DCN: makes use of optical devices to speed up communications. It can be either: i) all-optical DCN (i.e., with completely optical devices) or ii) hybrid optical DCN (i.e., both optical and Ethernet switches)
  2. Wireless DCN: deploys wireless infrastructure in order to enhance network performance, and may be: i) fully wireless DCN (i.e., only wireless devices) or ii) Hybrid DCN (i.e., both wireless and wired devices)

Figure 2.1 illustrates the taxonomy of current DCN architectures. In the following, we will detail each category and discuss their impact on Cloud computing performance.

2.2.2 Switch-centric DCN architectures overview

2.2.2.1 Tree-based DCN

The traditional DCN is typically based on a multi-root tree architecture. The latter is a three-tier topology composed by three layers of switches. The top level (i.e., root) represents the core layer, the middle level is the aggregation layer, while the bottom level is known as the access layer. The core devices are characterized by high capacities compared with aggregation and access switches. Typically, the core switches' uplinks connect the data center to the Internet. On the other hand, the

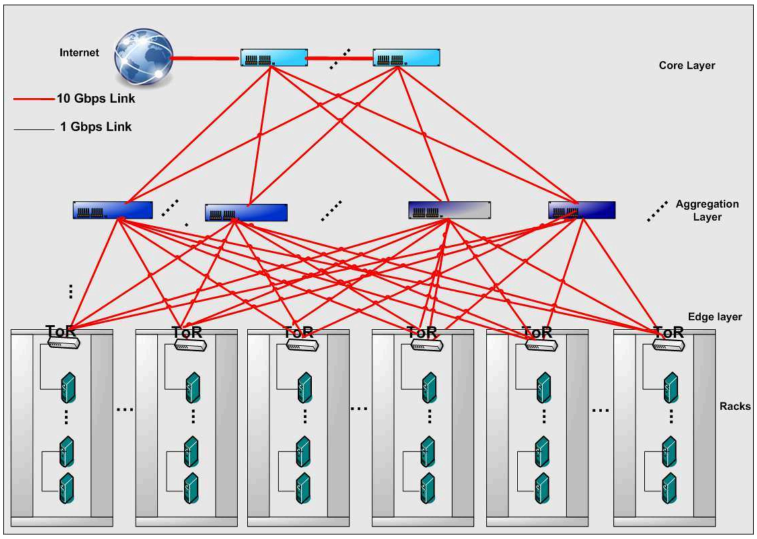


Figure 2.2: Traditional tree-based DCN architecture

access layer switches commonly use 1 Gbps downlink interfaces and 10 Gbps uplink interfaces, while aggregation switches provide 10 Gbps links. Access switches (i.e., ToRs) interconnect servers in the same rack. Aggregation layer allows the connection between access switches and the data forwarding. An illustration of tree-based DCN architecture is depicted in Figure 2.2.

Unfortunately, traditional DCNs struggle to resist to the increasing traffic demand. First, core switches are prone to bottlenecks issues as soon as the workloads reach the peak. Moreover, in such a DCN, several downlinks of a ToR switch share the same uplink which limits the available bandwidth. Second, DCN scalability strongly depends on the number of switch ports. Therefore, the unique way to scale this topology is to increase the number of network devices. However, this solutions results in high construction costs and energy consumption. Third, tree-based DCN suffers from serious resiliency problems. For instance, if a failure happens on some of the aggregation switches, then servers are likely to lose connection with others. In addition, resource utilization is not efficiently balanced. For all the aforementioned reasons, researchers put forward alternative DCN topologies.

2.2.2.2 Hierarchical DCN architecture

Hierarchical topology arranges the DCN components in multiple layers. The key insight behind this model is to reduce the congestion by minimizing the oversubscription in lower layer switches using the upper layer devices. In the literature, we find several hierarchic DCN examples, namely, CLOS, FatTree and VL2. Hereafter, we will describe each one of them.

CLOS-based DCN: is an advanced tree-based network architecture. It was, first, introduced by Charles Clos, from Bell Labs, in 1953 to create non-blocking multi-stage topologies, able to provide higher bandwidth than a single switch. Typically, CLOS-based DCNs come with three layers of switches: i) Access layer (ingress), composed of the ToRs switches, directly connected to servers in the rack, ii) Aggregation layer (middle), formed by aggregation switches referred as spines and connected to the ToRs, and ii) Core layer (egress), formed by core switches serving as edges to manage traffic in and out the DCN [26]. The CLOS network has been widely used to build modern IP fabrics, generally referred to as spine and leaf topologies. Accordingly, in this kind of DCN, commonly named folded-CLOS topology, the spine layer represents the aggregation switches (i.e., spines) while the leaf layer is composed of the ToR switches (i.e., leaves). The spine layer is responsible for interconnecting leafs. CLOS inhibits the transition of traffic through horizontal links (i.e., inside the same layer). Moreover, CLOS topology scales up the number of ports and makes possible huge connection using only a small number of switches. Indeed, augmenting the switches ports enhances the spine layer width and, hence, alleviates the network congestion. In general, each leaf switch is connected to all spines. In other words, the number of up (respectively down) ports of each ToR is equal to the number of spines (respectively leaves). Accordingly, in a DCN of n leaves and m spines, there are $n \times m$ wired links. The main reason behind this link redundancy is to enable multi-path routing and to mitigate oversubscription caused by the conventional link state OSPF routing protocol. In doing so, CLOS network provides multiple paths for the communication to be switched without being blocked.

CLOS architecture succeeds to ensure better scalability and path diversity than conventional tree-based DC topologies. Moreover, this design reduces bandwidth limitation in aggregation layer. However, this architecture requires homogeneous switches, and deploys huge number of links.

Fat-Tree DCN: is a special instance of CLOS-based DCN introduced by Al-Fares [27] in order to remedy the network bottleneck problem existing in the prior tree-based architectures. Specifically, Fat-Tree comes with a new way to interconnect commodity Ethernet switches. Typically, it is organized in k pods, where each pod contains two layers of $k/2$ switches. Each k-port switch in the lower layer is directly connected to $k/2$ hosts, and to $k/2$ of the k ports in the aggregation layer. Therefore, there is a total of $(k/2)^2$ k-port core switches, each one is connected to each port of the k pods. Accordingly, a fat-tree built with k-port switches supports $k^3/4$ hosts.

The main advantage of the Fat-Tree topology is its capability to deploy identical cheap switches, which alleviates the cost of designing DCN. Further, it guarantees equal number of links in different layers which inhibits communication blockage among servers. In addition, this design can importantly mitigate congestion effects thanks to the large number of redundant paths available between any two given communicating ToR switches. Nevertheless, Fat-Tree DCN suffers from complex connections and its scalability is closely dependent on the number of switch ports. Moreover, this structure is impacted by the possible low-layer devices failure which may entail the degradation of DCN performance.

This architecture has been improved by designing new structures based on a Fat-Tree model, namely, **ElasticTree** [28], **PortLand** [29] and **Diamond** [30]. The main advantage of such topologies is to reduce maintenance cost and enhance scalability by reducing the number of switch layers.

Valiant Load Balancing DCN architecture VLB is introduced in order to handle traffic variation and alleviate hotspots when random traffic transits through multi-paths. in the literature, we find, mainly, two kinds of VLB architectures. First, **VL2** is three-layer CLOS architecture introduced by Microsoft in [16]. Contrarily to Fat-Tree, VL2 resorts to connecting all servers through a virtual 2-layer Ethernet, located in the same LAN with servers. Moreover, VL2 implements VLB mechanism and OpenFlow to perform routing while enhancing load balancing. To forward data over multiple equal cost paths, it makes use of Equal-Cost Multi-Path (ECMP) protocol. VL2 architecture is characterized by its simple connection and does not require software or hardware modifications. Nevertheless, it still suffers from scalability issue and does not take into account reliability, since single node failure problem persists.
Second, **Monsoon** architecture [31], aims to alleviate over-subscription based on a 2-layer network that connects servers and a third layer for core switches/routers. Unfortunately, it is not compatible with the existing wired DCN architecture.

2.2.2.3 Flat DCN architecture

The main idea of the Flat switch-centric architectures is to flatten down the multiple switch layers to only two or one single layer, so as to simplify maintenance and resource management tasks. There are several topologies that are proposed for this kind of architecture. First, the authors of [32] conceive **FBFLY** architecture to build energy-aware DCN. Specifically, it considers power consumption proportionally to the traffic load, and so replaces the 40 Gbps links by several links with fewer capacity regarding the requested traffic in each scenario. **C-FBFLY** [33] is an improved version of FBFLY which makes use of the optical infrastructure in order to reduce cabling complexity while keeping the same control plane. Then, **FlaNet** [34] is also a 2-layer DCN architecture. Layer 1 includes a single n-port switch connecting n servers, whereas the second layer is recursively formed by n^2 1-layer FlatNet. In doing so, this architecture reduces the number of deployed links and switches by roughly $1/3$ compared to the classical 3-layer FatTree topology, while keeping the same performance level. Moreover, FlatNet guarantees fault-tolerance thanks to the 2-layer structure and ensures load balancing using the efficient routing protocols.

Discussion In conclusion, switch-centric architectures succeed to relatively enhance traffic load balancing. Most of these structures ensure multi-routing. Nevertheless, such a design brings up in general at least three layers of switches which strongly increases cabling complexity and limits, hence, network scalability. Moreover, the commodity switches commonly deployed in these architectures do not provide fault-tolerance compared to the high-level switches.

2.2.3 Server-centric DCN architectures overview

In general, these DCN architectures are conceived in a recursive way where a high-level structure is formed by several low-level structures connected in a specific manner. The key insight behind this design is to avoid the bottleneck of a single element failure and enhance network capacity.

The main server-centric DCN architectures found in the literature include **DCell** which is a recursive architecture built on switches and servers with multiple Network Interface Cards (NICs) [35]. The objective is to increase the scale of servers. Moreover, **BCube** is a recursive server-centric architecture [17], which makes use of on specific topological properties to ensure custom routing protocols. Finally, **CamCube** [36] is a free of switching DCN architecture, specifically modeled as a 3D DCN topology, where each server connects to exactly two servers in 3D directions.

Server-centric DCN architectures, leading on recursive network structures, succeed to alleviate the bottleneck in core layer switches thanks to redundant paths provided between servers. The entire DC fabric is built on servers while minimizing the set of deployed switches. Therefore, maintenance and management tasks become simpler. Moreover, network functions such as traffic aggregation, packet forwarding, etc, are delegated to servers. However, due to their recursive structure, server-centric structures significantly increase the number of servers, which would drastically increase the cabling complexity.

2.2.4 Enhanced DCN architectures overview

Despite the use of multi-gigabytes wired links and multi-port switches in order to balance the load, the aforementioned DCN architectures are still facing flexibility and congestion challenges. Recently, a promising solution has investigated the possibility of augmenting the wired infrastructure by novel networking techniques, to enhance the capacity of DCNs. In the literature, the augmentation of such a DCN can mainly be achieved using tow ways: i) **optical** or ii) **wireless** devices.

2.2.4.1 Optical DCN architecture

Optical Data Center Network (O-DCN) is a DCN architecture based on optical cabling and switching. Indeed, it has been found out that deploying such optical devices in DCs achieves a gain of 75% in IT power. Firstly, on-demand high-speed links can be easily established thanks to the flexibility of optical network compared to the traditional wired DCN. Secondly, optical devices are able to ensure high bandwidth over longer ranges, and avoid, hence, the cost required for cabling along large distances. Further, O-DCNs deploy optical switches with high-radix ports, characterized by a low temperature, so as to reduce refrigeration cost. O-DCN can be classified in two main classes: i) full optical DCN (all O-DCN) and ii) hybrid optical DCN (hybrid O-DCN), detailed hereafter.

Full O-DCN architectures: In such architectures, all the control and data planes devices are optical. The key idea behind this full optical deployment is to provide high-speed bandwidth in the DCN. In this regard, O-DCN makes use of several techniques. First, Optical Circuit Switching (OCS) [37] has been deployed in order to offer large bandwidth at the core layer. To do so, OCS DCN [38] proceeds to pre-configuring the static routing paths in the switches. Second, Optical Packet Switching (OPS), proposed in [37], provides on-demand bandwidth in the DCN. In [19], the **DOS** scalable DCN architecture has been propounded based on OPS technique. However, such an architecture suffers of low scalability. In addition, the Elastic Optical Network (**EON**) [18], is a kind of full O-DCN offering centralized on-demand flexibility in bandwidth switching.

Hybrid O-DCN architectures: Hybrid optical DCNs augment the wired DCNs by optical devices so that to provide extra bandwidth in an on-demand way by switching the connections in order to alleviate routing hop-counts. In doing so, hybrid O-DCNs succeed to minimize congestion effects on top of racks and to reduce traffic complexity by ensuring on-demand connections.

In this context, the authors of [39] introduced a novel Optical Switching Architecture (**OSA**) based on some techniques. Specifically, OSA makes use of a shortest path routing scheme and optical hop-to-hop switching in order to enable connectivity in DCN.

Moreover, **Helios** in [20], is a hybrid electrical-optical DCN, where each ToR is connected simultaneously to an electrical and an optical network. While electrical network is a Fat-tree hierarchical structure, the optical one maintains a single optical connection on each ToR, with unlimited capacity. Helios deploys mirrors on a micro-electro mechanical system to route the optical signals so as to alleviate traffic congestion at core level.

An additional example of hybrid O-DCN is **c-Through** [40], a platform that includes a control and a data plane. The control plane measures an estimation of inter-rack traffic demands, then it dynamically calibrates circuits in a way that accommodates the new incoming flows. On the other side, the data plane isolates the electrical network from the optical one, and dynamically switches traffic from servers or ToRs onto the the circuit or packet path. c-Through favors the use of optical paths as long as they are available, compared to the electrical routes. Nevertheless, it is worth pointing out that both of Helios and c-Through architectures fail to alleviate routing overheads.

FireFly is a wireless optical DCN architecture based on Free-Space Optics (FSO) [41]. The main advantage of such a design is that it provides a high data rate ($\approx$ tens of Gbps) for long communication range while using low transmission power without interference. Specifically, servers in different racks communicate with each other using FSO reflected on ceiling mirrors.

Discussion In conclusion, enhancing DCN with optical technique succeed to satisfy many Cloud computing requirements. Particularly, it provides high-speed traffic with low power consumption. Optical links alleviate the overhead compared to electric links. The aforementioned research optical approaches offer flexible switching solutions in order to make easy the bandwidth management for on-demand Cloud services. However, this designs still suffer from several limitations. First, O-DCN induces switching overhead. In fact, it requires the deployment of some modulation schemes in order to properly adjust bandwidth while switching connections, which is a challenging task. Second, O-DCN can not be deployed in large-scale environments so far because of the high cost of optical transceivers and their long latency. Third, a significant reconfiguration latency of roughly 10 ms is induced by O-DCN which would affect applications QoS, such as online services.

2.2.4.2 Wireless DCN architecture

To address the challenges of both wired and optical DCN in terms of cabling complexity, deployment cost, scalability, and so on, Wireless DCN (W-DCN) has been recently explored. W-DCN architecture deploys wireless antennas, operating in the 60 GHz frequency band, to connect pairs of ToR switches. In doing so, the wired infrastructure is augmented with inter-rack wireless links. The main insight behind this approach is to investigate the high data transfer rate of this new emerging technique, that can reach 7 Gbps, in order to enhance DCN performance. Actually, a 60 GHz wireless link makes use of the physical beamforming technique so that the transmitted signal is concentrated in a specific direction enhancing while mitigating interference. The related wireless DCN architectures found in the literature could be classified to: i) hybrid W-DCN and ii) full W-DCN. Hereafter, we will detail the most relevant wireless DCN architectures.

Hybrid wireless DCN architectures: In such an architecture, both wired and wireless infrastructures are used in the same DCN. Wireless augmentation of DCN has been first explored by the authors of [42] in order to reduce cabling complexity in the wired DCN while enhancing network flexibility. The main idea behind their design is to replace some of wired bottleneck links by wireless connections operating in the 60 GHz range. Besides, [43] designs a wireless DCN based on IEEE 802.5.3c standard [44] in the wireless 60 GHz communications. To study the feasibility of such technique in DCN, the authors emulate three-tier and Fat-Tree architectures with wireless links. To do so, they propose node placement algorithms to assign nodes to racks.

Later on, **Flyway-based** DCN architecture [11] [45] has been propounded in order to alleviate congestion on hotspot links in the VL2 architecture [16]. However, Flyway links are created on-demand in the DCN as long as there is congestion on the ToR and struggle to meet all the challenges of DCN such as scalability, high traffic load and interference.

The authors of [8] have proposed a hybrid wired/wireless DCN architecture where each ToR, considered as a Wireless Transmission Unit (WTU), is equipped with a set of wireless 60 GHz radios. This hybrid architecture investigates the use of wireless infrastructure in order to reduce the congestion level of congested nodes and to handle unbalanced traffic demands in DCN.

In [10], the authors envision a hybrid Ethernet/wireless tree-layered DCN architecture. Congestion on core layer is alleviated by deploying 60 GHz wireless antennas on top of racks, without needing to rearrange servers in the same rack.

To further enhance the DCN performance, some research work papers have investigated the use of beamforming technique while designing hybrid DCN architectures. Particularly, 3D beamforming has been presented in [46] and [47] in order to boost the transmission range and 60 GHz spectrum reuse in DCNs. Basically, the enhanced design sets up indirect LOS path by making use of ceiling reflectors. These latter enable the interconnection of wireless antennas that are not placed in the same transmission range. Typically, the horn antenna placed on each sending rack radiates the signal in some points on the reflector, and the latter transmits the signal to the receiver. In doing so, obstacles are eliminated and racks could communicate directly in one hop. While this 3D beamforming architecture significantly extends wireless coverage distance, it requires the absence of obstacles between the top of rack/container and the ceiling which is not guaranteed in real DC environments.

The authors of [48] investigate the use of steered-beam antennas in order to build a robust wireless **crossbar** switch-centric DCN architecture. In such a design, wired cabling is used only for intra-rack links or to interconnect racks within the same row. On the other hand, wireless steered-beam antennas are deployed on adjacent ToRs while constituting a wireless crossbar so that cabling task is simplified and installation cost is reduced.

Angora architecture recently proposed in [7] propounds a robust wireless topology for the control plane while data is completely transiting over wired infrastructure. To do so, 3D beamforming radios are deployed on racks based on Kautz graphs, so that network latency is reduced by minimizing the path length between communicating racks. Moreover, Angora alleviates inter-flow interference by statically calibrating the directions of the deplyed horn/array antennas. Unfortunately, the static 3D direction of antennas may strongly limits the usage of spectrum.

In [49], a **spherical mesh** is a wireless DCN where racks within the same transmission range are regrouped into a spherical unit. The main idea is to take profit of the geometric characteristics of the spheres to eliminate link congestion by placing antennas over them. Moreover, the spherical mesh DCN reduces the network diameter by dividing the DCN into several units.

RUSH DCN architecture is proposed in [50], which is a hybrid DCN based on the common three-layer tree topology. In RUSH, each ToR is equipped by only one directional 60 GHz antenna and wireless inter-rack links are used to minimize congestion. For that end, the authors propose a scheduling framework to jointly route flows and schedule wireless antennas.

In [51], **Diamond** DCN architecture is improved by deploying 3D wireless rings. Unlike common hybrid designs, Diamond is a hybrid wired/wireless DCN where all links between servers are wireless, whereas links connecting servers to ToRs or connecting ToRs are wired. The rings consist in regular polygons which are constructed by racks and metal reflectors, while the layers contain the servers inside racks belonging to the same level. The main reason behind the use of 3D Ring Reflection Spaces (RRSs) is their low-cost and their ability to provide wireless links by multi-reflection of signals over metal. Diamond feasibility has been studied based on a real testbed.

VLCcube is a hybrid DCN architecture which is propounded in [52]. It is an augmented Fat-Tree structure that specifically organizes all racks into a wireless Torus structure while making use of the Visible Light Communications (VLC) technique to generate high-speed links. In doing so, all racks are connected based on VLC links. VLC is a promising solution that guarantees low cost and important bandwidth. Moreover, VLC links do not require mechanical or electronic control.

Full wireless DCN architectures: A completely wireless DCN architecture has been propounded in [21], based on a Cayley graph, thereby named Cayley Data Center structure (**Cayley DC**). The servers are grouped into cylindrical racks. Each one is composed by 5 levels named stories. A story consists of 20 containers of servers. Racks are attached to densely wireless connected mesh topology with the aim of maximizing the number of active wireless links. Specifically, the Cayley DC uses wireless links not only for inter-rack communications but also inside racks, thanks to the mesh structure. In order to alleviate interference effects, this strategy makes use of beamforming technique with fixed-direction antennas.

Discussion To summarize, most of the relevant research work published in the recent years approves the feasibility and the efficiency of deploying 60 GHz wireless technology as an extension of conventional wired DCN architectures. Hybrid wireless/wired DCN have proven a significant capability to enhance network performance and to address the major data center issues, namely scalability, flexibility, and cabling complexity.

2.3 Comparison between DCN architectures

In this section, we will present a qualitative comparison between the reviewed DCN architectures while considering some specific criteria: scalability, bandwidth, cabling complexity, deployment cost and fault tolerance. Scalability refers to the ability of the proposed architecture to easily scale and deploy more devices. Bandwidth represents the proportion of available bandwidth between servers and switches, while cabling complexity refers to the multitude of cables in the DCN induced by link redundancy. The overheads and the cost of deployment in DCN are also crucial factors that refer to the number of switches and links and their corresponding construction and deployment cost. Finally, fault-tolerance defines the ability of the designed architecture to deal with switch and link failures.

Table 2.1 illustrates a comparison between different DCN architectures based on the aforementioned aspects.

Table 2.1: Summary and analysis of DCN architectures

Architecture	technique	Scale	Bandwidth	Scalability	Cabling complexity	Cost	Fault-tolerance
Tree-based	wired	small	low	bad	high	high	bad
CLOS [26]	wired	medium	medium	medium	high	high	medium
FatTree [27]	wired	medium	medium	medium	high	high	medium
ElasticTree [28]	wired	medium	medium	medium	high	high	medium
PortLand [29]	wired	medium	quite high	medium	high	high	good
Diamond [30]	wired	medium	high	medium	high	Low	medium
VL2 [16]	wired	Large	quite high	medium	high	high	medium
Monsoon [31]	wired	Large	quite high	medium	high	high	medium
FBFLY [32]	wired	Large	high	medium	high	high	medium
FlaNet [34]	wired	Large	high	Low	high	high	medium
C-FBFLY [33]	wired	Large	high	Low	high	high	medium
DCell [35]	wired	Large	high	good	high	medium	good
BCube [17]	wired	small	very high	good	medium	medium	very good
CamCube [36]	wired	Large	high	good	very high	high	good
FiConn [53]	wired	Large	high	good	medium	Medium	good
O-DCN [40] [20] [41]	optical	small	very high	medium	high	high	bad
Flyway-based [45] [11]	60 GHz/Ethernet	medium	very high	good	medium	medium	good
Wireless Fat-Tree [43]	60 GHz/Ethernet	medium	very high	good	medium	medium	medium
Hybrid DCN [8]	60 GHz/Ethernet	medium	very high	good	medium	medium	medium
Hybrid DCN [10]	60 GHz/Ethernet	medium	very high	good	medium	medium	medium
3D Beamforming [46] [47]	3D Beamforming	medium	very high	good	medium	high	medium
Wireless crossbar [48]	60 GHz	medium	very high	good	medium	high	medium
Cayley DC [21]	60 GHz	medium	very high	good	medium	high	good
Angora [7]	60 GHz/Ethernet	medium	very high	good	medium	high	medium
Spherical mesh [49]	60 GHz/Ethernet	medium	very high	good	high	high	medium
RUSH [50]	60 GHz/Ethernet	medium	very high	good	medium	high	medium
3D Diamond [51]	3D Beamforming/wired	medium	very high	good	medium	high	medium
VLCcube [52]	VLC	medium	very high	good	medium	high	medium

2.4 Proposed HDCN architecture

In this thesis, we envision a Hybrid (wireless/wired) Data Center Network (HDCN) architecture built over a three-stage CLOS topology. Indeed, as explained in Section. 2.2, CLOS-based architecture has been widely considered in modern DCs and has proven a high performance and resiliency. To mimic a real data center environment, our CLOS-based HDCN architecture follows the CISCO's Massively Data Center (MSDC) model [22]. In fact, MSDC is a promising framework capable of supporting huge volume of traffic. To augment the wired infrastructure in HDCN by wireless links, we make use of 60 GHz wireless technology. In doing so, traffic can be forwarded over wireless and/or wired links which will alleviate the congestion load and hence improve the network performance.

In this section, we will first highlight the main properties of MSDC model. Second, we will focus on the wireless infrastructure in the HDCN by presenting the: i) 60 GHz technology, ii) IEEE 802.11ad standard and iii) deployed beamforming mechanism.

2.4.1 HDCN architecture based on MSDC model

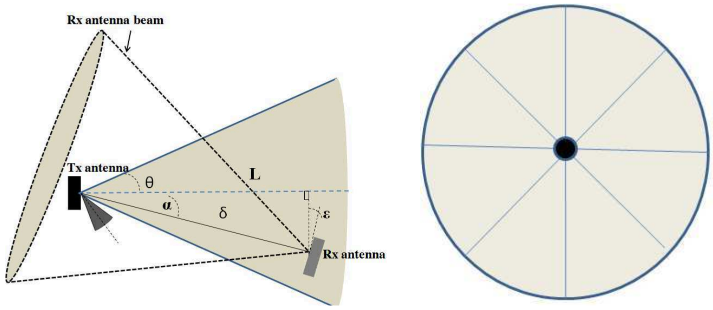
CISCO's Massively Scalable Data Center (MSDC) is a framework model that has been widely used by data center architects to build flexible DCs supporting applications distributed across thousands of servers. Typically, MSDC is built based on a CLOS-based topology with a short spine layer serving as the aggregation switches, and a long leaf layer serving as the access layer. Specifically, a three-stage CLOS MSDC architecture using 32 port switches, and can thus connect up to 8192 servers. Based on the CISCO's MSDC reference [22], our HDCN architecture follows a three-stage CLOS topology formed by: i) spine layer using Nexus 7000 switches, and ii) leaf layer deploying Nexus 3000 platform. Each leaf connects to all spines. In doing so, our MSDC-based HDCN network provides multiple paths for inter-rack communications between servers. To leverage the multiple paths available between leaf and spine switches, MSDC data center deploys both OSPF routing and Equal Cost Multipathing (ECMP) protocols. ECMP maximizes the load balancing of wired links' usage by dividing the traffic through multiple equal cost routes. Hereafter, we will detail the load-balancing ECMP mechanism used in our HDCN.

2.4.1.1 ECMP protocol

ECMP [22] is the most commonly used protocol in today's data centers, for the traffic load balancing across redundant shortest routing paths. The main idea of ECMP is to divide the traffic through multiple equal cost routes. Basically, this technique is a selection tool that finds the convenient route for each transmitted packet and this by choosing the next hop from the computed OSPF routes. Mainly, two modes of load balancing are associated to ECMP: i) Per packet mode, where the packets of the same flow may have different routes, and ii) Per flow mode, where the packets of the same flow are forwarded to the same next-hop, ensuring the ordered arrival of packets in TCP mode. In this thesis, we generate traffic, in HDCN, based on User Datagram Protocol (UDP). Consequently, based on ECMP RFC [54], ECMP activates i) the mode per-packet to maximize the load balancing and ii) Round Robin scheduler to select the next hop (outgoing interface) for each packet.

2.4.2 60 GHz technology in HDCN

As in prior work [6] [45] [42] [21], we propose in this thesis to deploy 60 GHz wireless technique in order to enhance our hybrid DCN architecture. Specifically, wireless infrastructure in our HDCN is based on IEEE 802.11ad. This standard, presented by the working group TGad as the enabler of next generation Multi-Gbps WiFi, takes advantages of available spectrum in the unlicensed 57-66 GHz band. It offers 4 orthogonal physical channels whose center frequencies are respectively fixed at 58.32, 60.48, 62.64 and 64.8 GHz. The capacity of each wireless channel reaches 6.7 Gbps over a short range. Consequently, the whole network of a data center can be seen as Personal Basic Service Set (PBSS). Indeed, PBSS is IEEE 802.11ad wireless LAN in which stations communicate directly with each other (i.e., Ad hoc network, no need of access point) [23]. Note that each node in PBSS is denoted by Directional Multi-Gigabit Station (DMG-STA). The latter is defined in the standard as a station operating at a frequency above 45 GHz and can support a throughput greater or equal to 1 Gbps. In PBSS network, one DMG-STA must assume the role of controller and is denoted by PBSS Control Point (PCP). It ensures the QoS traffic scheduling, resource allocation, control admission, association/disassociation, etc. In other words, the PCP has a global view of nodes in PBSS. PCP is a global controller responsible for the i) management of the Hybrid DCN and ii) optimization of the resource usage and flows forwarding. It is worth noting that the communication between the PCP and all the DMG-STA in PBSS should be ensured over wireless network. However, some WTU deployed over DMG-STA cannot reach the PCP in wireless one-hop. In our architecture, we propose that communications between the PCP and WTUs will be supported by the wired infrastructure (e.g., Ethernet, OpenFlow, etc.). In doing so, we can see our architecture as a Software Defined Network. In fact, the control plane is centralized in the PCP and WTUs support only the data plane (i.e., transmission of frames). IEEE 802.11ad standard defines three frame classes. In our Hybrid DCN (i.e., PBSS), we leverage the frames of Class 1. The latter contains three frame types: i) control frames, ii) data frames and iii) management frames. Concerning the control frames, we only make use of ACK frames. They are transmitted over a single carrier modulation by setting the Modulation and Coding Scheme (MCS) to 0. The latter corresponds to DBPSK modulation, code rate is $\frac{1}{2}$, data rate is 27.5 Mbps and receiver sensitivity is -78 dBm. On the other hand, data frames are transmitted over Orthogonal Frequency-Division Multiplexing (OFDM) modulation by setting MCS to 24. The latter corresponds to 64-QAM modulation, code rate is $\frac{13}{16}$, data rate is 6756.75 Mbps (i.e., maximum data rate) and receiver sensitivity is -47 dBm.



(a) Spherical coordinate system

(b) Beams

Figure 2.3: Switched-beam antenna model

Finally, the management frames are transmitted over the wired infrastructure.

Based on this specification, we propose to deploy at each ToR a WTU composed of a set of 4 directional transceivers/antennas. Each transceiver is, hence, assigned to one wireless channel. Note that the 4 transceivers in WTU are independent, due channel orthogonality, and can be simultaneously exploited. In doing so, any rack in the data center can communicate over the wired ports (i.e., ToR) and/or using wireless channels. It is worth pointing out that the wireless 60 GHz communication is faced to several challenges due to the free space propagation loss. The latter is due to the low power density, and results in a short transmission range. Moreover, wireless links are prone to interfere in HDCN environment which deeply affects transmission stability. To address these limitations, we explore in this thesis beamforming technique so that to minimize the propagation loss and increase coverage distance.

2.4.3 Beamforming technique in HDCN

The beamforming is a physical layer technique that concentrates transmission power in a specific direction (i.e., beam), so that the link rate is enhanced. Unlike omni-directional antennas radiating signal in uniform way (circle), smart directional transceivers are capable of transmitting signal in one single beam (angle) by targeting only the direction of the destination. Typically, a directional antenna is in general composed by: i) an array of antenna elements (beams) and ii) a signal processor adjusting the radiation of the latter.

Mainly, current 60 GHz beamforming antennas are available either as horn antennas [6], phased-array antennas [55] or switched-beam antennas [7]. While the phased-array transceivers are steerable devices that appropriately steer each beam at the desired target direction, horn antennas are in general used for fixed links, in long range outdoor environments. Recent researches [7] [47] claim that both array and horn antennas require a mechanical rotation mechanism at each single communication to adjust the beam direction. This frequent antenna rotation induces an extra delay estimated to equal 50 ns for array antennas and to range from 0.01 to 1 second for horn antennae [47].

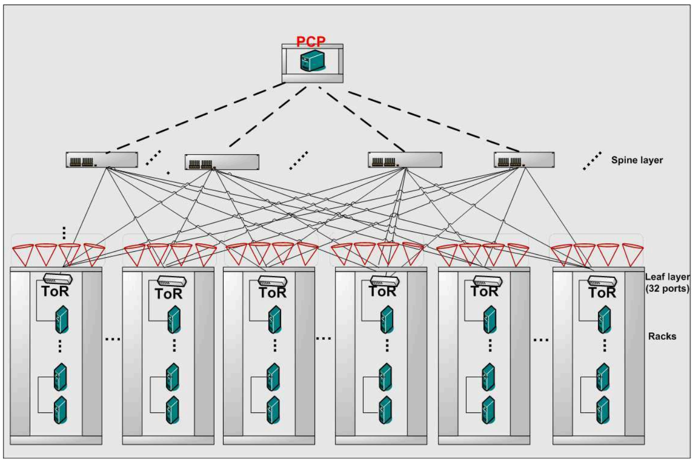


Figure 2.4: Hybrid Cisco MSDC architecture of a data center network

Based on these observations and as recommended by [7], we deploy, in this thesis, switched beam antennas to avoid performance degradation. In fact, such devices have been considered to be less complex than the other smart radios and are cheaply implemented. As depicted in Figure 2.3(b), a switched beam antennae is characterized by an array of N beams (i.e., sectors). Each one covers an angle of $2\Pi/N$. Accordingly, the transmitting antenna switches to (i.e., selects) the beam achieving the highest gain while covering the destination. The receiving antenna senses the signal on all the sectors and exploits only the one achieving the maximum gain. The signal coming from potential interfering antennas is either not received or significantly weak.

We assume the geometric signal propagation model [21] based on a spherical coordinate system with origin the transmitting antenna as shown in Figure 2.3(a). The receiver antenna is characterized by radius δ, azimuth θ as shown in Figure 2.3(a). Note that we assume 2D beamforming and hence elevation is equal to 0.

Our HDCN architecture is illustrated in Figure 2.4.

2.5 Conclusion

In this chapter, we provided an overview of data center network architectures. First, we proposed a taxonomy classifying the relevant DCN structures into three main classes: i) switch-centric, ii) server-centric and iii) enhanced DCN architectures. We deeply analyzed the key properties of each class. Afterwards, we provided a qualitative comparison study between the different DCN architectures. Finally, we presented our chosen hybrid DCN architecture based on i) Cisco's MSDC framework and ii) wireless 60 GHz technique. In the next chapter, we will present a detailed review on the most relevant research strategies in the literature tackling wireless resource allocation problem for both one-hop and multi-hop communications in HDCN.

Chapter 3

Routing and Wireless Resource Allocation in Hybrid Data Center Networks: Overview

Contents

3.1 Introduction

Routing and resource allocation are key challenges in hybrid data center networks. Ensuring an efficient management of wireless and wired infrastructure in the HDCN, for both one-hop and multi-hop communications, is primordial to guarantee a high performance network. For **one-hop inter-rack communications**, where the sending and receiving racks are placed in the same wireless transmission range, the objective is to find efficient algorithms for wireless channel allocation in HDCN while minimizing the congestion level. Several recent research approaches [6] [46] have explored the feasibility of deploying wireless links in HDCN based on practical testbeds, but only few studies have been conducted to perform channel allocation.

On the other hand, the **multi-hop inter-rack communications** require efficient mechanisms to jointly route and allocate channels for the communication flows, while enhancing network performance. The objective is to compute for each flow, the hybrid (i.e., wireless and/or wired) routing path. In this regard, the joint routing and wireless channel allocation problem in HDCN can be addressed either in an online or a batch way. In the online mode, inter-rack communication flows are sequentially processed in order to find the hybrid routing path for each single flow request. Few research works have been proposed to deal with this issue. However, even if the online approaches guarantee an optimized hybrid routing path for each single flow request, they fail to ensure an optimized use of the wireless and wired resources in the HDCN. Indeed, the arrival order closely impacts the HDCN performance. Therefore, a few recent researches have investigated the Joint Batch Routing and Channel Assignment problem (`JBRC`) in HDCN, to handle the batched arrivals of communication flows. Their objective is to find, for each batch of flows, the corresponding hybrid routing paths.

In this chapter, we will review the different routing and wireless resource allocation strategies in HDCN. For the sake of completeness, we first give a brief description of the above problems and their challenges in HDCN. Then, in the second section, we will give an in-depth overview of the wireless channel allocation approaches dealing with one-hop inter-rack communications in HDCN. Next, we introduce the major joint online routing and channel allocation strategies for multi-hop communications in HDCN. Afterwards, the main joint batch routing and channel allocation algorithms dealing with the batched arrival of inter-rack flows are detailed in section 3.6. Then, we will present a qualitative comparison between the different related resource allocation and routing strategies in HDCN. Finally, we summarize this chapter.

3.2 Routing and wireless channel allocation problematic in HDCN

Intra-DCN communication flows can be either within the same rack (i.e., intra-rack) or between servers from different racks (i.e., inter-rack). In the context of HDCN, augmented with inter-rack wireless links to alleviate over-subscription, researches mainly focus on inter-rack communications. An inter-rack communication request is characterized by: i) a sending rack, ii) a receiving rack, and iii) a traffic flow to be transmitted between them. We recall that each top of rack deploys: i) a Wireless Transmission Unit (WTU) which is equipped with 4 IEEE 802.11ad transceivers/antennas and ii) a wired Ethernet switch. One of the key features of HDCN is its ability to efficiently: i) allocate wireless/wired resources and ii) route flows, for on-demand intra-DCN communications. In this respect, an efficient wireless channel allocation strategy is required so that both wireless and wired links in HDCN are judiciously allocated to ongoing communications while minimizing the end-to-end delay. The main objective of wireless channel allocation problem in HDCN is to maximize the proportion of intra-data center communication requests transiting over the wireless infrastructure. In doing so, the end-to-end delay of communications and the congestion of wired infrastructure are minimized, and hence, the total throughput in the HDCN is maximized. Formally, the main purpose is to satisfy each communication flow requirements, in terms of bandwidth, while minimizing congestion and alleviating interference between ongoing wireless links. It is worth noting that wireless channel allocation problem in HDCN has proven to be `NP-hard` [8], due to interference constraint and the limited number of wireless channels.

Furthermore, the hybrid DCN architecture is faced to the short range limitation of the 60 GHz frequency band. Consequently, inter-rack communications can not always be ensured in a single hop. To deal with this challenge, a few recent approaches have addressed the joint routing and channel allocation problem. The key insight of these methods is to jointly harness wireless and wired interfaces to enhance the data center network capabilities in term of bandwidth. In doing so, the end-to-end delay and the congestion of wired infrastructure are minimized. Formally, assuming an inter-rack communication flow from a source to a destination, the objective is to compute the best hybrid (i.e., formed by wireless and/or wired links) routing path while assigning wireless channels along links. The complexity of such a problem resides in the fact that channel allocation along the routing path should consider both: i) the available bandwidth on each link and ii) the level of wireless interference among intra-flow and inter-flow links, so that the end-to-end delay can be reduced.

3.2.1 Routing and wireless channel assignment challenges in HDCN

The routing and wireless channel allocation problem, for both one-hop and multi-hop communications, is extremely challenging for many reasons:

- **Arrival of inter-rack communication flows:** The inter-rack communication flows arrive to the HDCN is in dynamic way. Several research works [41] [52] model the arrival time of such requests as a Poisson process distribution with an inter-arrival λ_A. Each communication flow is characterized by its: i) source rack, ii) destination rack, iii) arrival time and iv) volume of traffic. According to the distance between the sending and receiving racks, the flow can be transmitted either in one single hop or multiple hops. Communication flows are not predictable in advance, as they dynamically arrive to the data center and transmit a random traffic. Therefore, their processing is extremely hard since the traffic in real DCN environment is very unbalanced, while the response time should be minimized as long as possible.

- **Unbalanced traffic demands in HDCN**: One main specificity of traffic demands in data center applications is its unbalanced criteria. That makes, unfortunately, the resource management harder in HDCN. Indeed, traffic unbalancing entails traffic concentration problem. For instance, recent traffic statistics obtained from real DC applications such as map-reduce usually concentrate their traffic in only a few hot nodes [8]. The latter induces bottlenecks and further delay the completion time of ongoing communications. Moreover, the random distribution of hot nodes makes it challenging to properly add new wireless links and alleviate ToRs congestion.

- **Wireless interference constraints:** Only 4 wireless channels are available for each deployed antenna operating with the IEEE 802.11ad standard. Although those channels are orthogonal and can be used simultaneously by the same rack, the traffic density in HDCN is likely to induce interference problem. In fact, wireless links that are in the same interference area can not make use of the same wireless channel at the same time. Otherwise, collisions will occur in the medium and consequently the QoS will be deteriorated. Therefore, wireless links should be appropriately established between ToRs in such a way that avoids interferences between wireless channels. It is worth noting that for the case of joint routing and channel assignment problem, two kinds of interference have to be considered. Actually, collisions may occur between links of the same routing path supporting the flow (intra-flow interference) as well as between links from different paths (i.e., flows) (inter-flow interference).

- **Limited resources:** Both wireless and wired resources in HDCN are limited. In fact, a single ToR switch is shared by all the servers of the same rack. Therefore, if a rack participates to many communications simultaneously, then the wired uplinks and downlinks of the ToR switch will be strongly congested. Moreover, only 4 wireless channels are available on each ToR. DC provider must optimize the allocation of wireless antennas and channels in aim to maximize the network performance.

- **Congestion on ToR switches:** ToR switches suffer from high congestion level. Hotspot links are consequently emerging in the HDCN and oversubscription has to be alleviated by properly allocating non-interfering wireless links.

- **Decision making** The routing and wireless resource allocation in hybrid DCN can be performed either in a centralized or in a distributed way. In the centralized scheduling [6] [8] [9] [50], a single centralized controller in the DCN infrastructure is responsible for both the traffic collection and the decision processing. Specifically, having a global view on the available resources in the HDCN, the centralized controller makes an optimal decision about the routing and resource allocation for the incoming traffic requests. Despite the advantages of such an approach, the centralized controller may be a bottleneck and a single point of failure. In the distributed decision [10] [52] [56], the routing and channel allocation decision is performed by different nodes in the DCN. Each entity has a local view of the DCN and is able to resolve a part of the decision problem. Then, all the decision-makers coordinate together to find the global best solution. However, it is straightforward to see that there is no guarantee of the optimality.

3.2.2 Routing and wireless channel assignment criteria in HDCN

Both routing and channel allocation mechanisms should take into account several criteria related to the network performance and to the infrastructure provider revenue. Typically, the most relevant criteria considered in the context of HDCN consist in:

- **Network throughput:** The main objective of Cloud data center providers is to enhance network performance by maximizing the throughput of applications. Typically, the total network throughput corresponds to the cumulative transmission throughput of the traffic carried through the hybrid DCN.

- **Traffic volume:** Obviously, the total throughput is an important metric for wireless resource allocation problem. However, it is not sufficient in the context of HDCN. In fact, racks requesting a higher amount of traffic usually requires longer time to carry their transmission due to the bandwidth limitation. Thus, they are likely to further increase the global completion delay. Accordingly, traffic volume of communication flows strongly impact the HDCN performance and it is in general considered in related work such as [8].

- **Total network Delay:** Estimating the network delay of each communication is mandatory to ensure a good DCN performance. In fact, a transmission with a high network delay that is caused by a congestion or a long communication path, may deteriorate DCN QoS. Thus, it is judicious to deploy wireless links in order to reduce the latency. The total delay of the network defines the cumulative transmission delay of all the finished communications in the network.

- **Spectrum Spatial Reuse:** Enhancing the spectrum reuse in very important to ensure an optimal use of the wireless infrastructure in the HDCN. The Spectrum Spatial Reuse (SSR) of a channel corresponds to the number of wireless communications which are simultaneously using the same wireless channel. Note that four wireless channels are available for IEEE 802.11ad.

- **Link distance:** Corresponds to the distance between the two communicating servers or racks. Actually, each rack in the HDCN is defined with its geographical position, and accordingly the hop distance, between the source and the destination of each transmission, can be defined. The latter strongly impacts the network utility. In fact, flows with longer paths usually induce higher transmission latency and thus increase the load of switches. Therefore, it is usually recommended to assign such flows to wireless links so as to alleviate congestion. However, this solution may incur a higher potential interference on wireless links. Further, the distance between two communicating racks decides whether a single-hop or multi-hop communication has to be established. Authors of [10] consider this parameter to define their objective network function.

- **Interference rate:** The set of interfering links on an interface is a decisive parameter that impacts the quality of the link. In fact, the larger is the number of conflict edges, the higher the latency is, which may aggravate network performance.

- **Link Cost:** The link cost is a crucial metric that deeply impacts the HDCN efficiency. In fact, it is an incarnation of the link congestion level, and the transmission delay. It is judicious to allocate wireless and/or wired links with lowest costs. It is worth pointing out that the cost of a link incarnates the transmission delay of its residual (wireless or wired) traffic and the resulting re-transmission delays (wireless) caused by/on interfering links.

- **Wireless requests use:** To evaluate the ability of the wireless resource allocation and routing strategies to efficiently carry incoming communications while minimizing congestion, it is important to evaluate the rate of requests that are assigned to wireless channels. In doing so, the efficiency of decision algorithms in allocating resources is gauged.

3.3 Wireless Channel Allocation strategies for one-hop communications in HDCN

We investigate, in this section, the existing wireless channel assignment approaches proposed for one-hop communications in hybrid DCNs. These strategies deal with wireless channel allocation problem for communications between two racks within the same transmission range. They can be classified into two main classes: i) omni-directional antennas based strategies and ii) Beamforming based strategies.

Hereafter, we will, first, discuss the main specificity distinguishing the wireless channel allocation problem in HDCN from that in classical wireless networks. Next, we will discuss in details the main proposals found in the literature.

3.3.1 Channel allocation problem in wireless networks

A rich panoply of researches have been studying the problem of channel allocation for wireless and cellular networks in the last decade. For instance, several approaches have been recently proposed to deal with this issue in the context of cellular mobile networks [57] [58]. The main challenge in such a problem lies in ensuring an efficient utilization of channels while considering interference constraints. To do so, several heuristic techniques, such as genetic algorithm, tabu-search and simulated annealing, have been used to tackle this NP-hard problem. In the other hand, wireless spectrum allocation has been addressed in the context of IEEE 802.11 WLANs so as to judiciously assign channels among Access Points [59]. In addition, this issue was tackled for sensor networks as in [60] [61], by proposing efficient protocols for multi-channel communications for IEEE 802.15.4 WSN while minimizing interference. It is worth pointing out that despite the performance of such proposed channel allocation solutions in the context of sensor or cellular networks, they can not, unfortunately, be applied for HDCN. Actually, we harness in hybrid data centers both wireless and wired resources. In other words, not only interference constraints are taken into account, but also the waiting delay on IP queues. Thus, both wired and wireless interfaces are jointly considered during the allocation process, in such a way that maximizes the amount of traffic transiting over wireless links, so that congestion on ToRs is alleviated and the throughput is enhanced.

3.3.2 Omni-directional antennas based strategies

- In [8], the authors propose a hybrid Ethernet/wireless DCN architecture to handle the limitations of Ethernet based DCN architectures and boost network performance. The wireless channel allocation problem is formulated as an optimization problem where the objective is to maximize the total throughput while satisfying interference constraints. In this context, a Genetic heuristic-based approach, names `Genetic-HDCN` is put forward to solve the optimization problem while handling traffic demands. Formally, each individual is defined as the channel allocation scheme associating to each ongoing transmission link the proper channel. A feasible individual is a channel allocation scheme satisfying interference constraints. The individual candidates that have the highest total throughput are selected. Moreover, `Genetic-HDCN` makes use of improved crossover and mutation operators. However, the initial population of solutions is randomly generated by the Genetic algorithm which may notably affect the quality of the final solution. Furthermore, the proposed solution is heuristic based and hence does not guarantee an optimal or near-to-optimal solution. Moreover, it is well known that Genetic heuristic struggles to converge for some problem instances. According to the simulation results, `Genetic-HDCN` strategy improves the HDCN performance

Algorithm 1: `Genetic-HDCN` pseudo-algorithm

```
Inputs: $m\ individuals\ X = \{X_1; X_2; ...; X_m\}$
Output: $optimal\ solution\ Y = \{Y_1; Y_2; ...; Y_m\}$
$Y \leftarrow \emptyset$
while There is evolution for one generation do
    $X_1 \leftarrow$ Selection(X)
    Divide the individuals in $X_1$ into pairs randomly; denote the set of pairs as $X_p$
    Apply Crossover operator
    Apply improved Mutation operator
    $Y \leftarrow$ Individual with best fitness
```

compared with the conventional `Wired-DCN` approach. `Genetic-HDCN` pseudo-code is summarized in Algorithm 1.

- In [9], the authors deal with the dynamic channel scheduling in wireless DCN. This approach assigns a weight to each edge. The latter corresponds to the transmission delay and reflects the level of the link contribution to the global DCN performance. The wireless transmission scheduling is formulated as an optimization problem. Then, a 0.5-approximation algorithm, `Approximation-HDCN`, is propounded to find an optimized channel allocation solution. This algorithm is based on a relaxation-rounding technique dealing with the relaxation of the the original integer optimization problem. To prove its efficiency, the authors compare the performance of their approximation algorithm to their previous proposal `Genetic-HDCN` [8]. Simulation results show that this approach outperforms the heuristic-based solution, and both strategies improve the performance compared with `Wired-DCN`. Unfortunately, this paper assumes omni-directional antennas deployed in top of racks, which maximizes the interference effects in the HDCN.

- The authors of [10] consider each ToR as a Wireless Transmission Unit (WTU). They formulate, first, the one-hop channel allocation problem in HDCN as an optimization scheme while maximizing the utility of the network. Such a utility is defined as the product of the traffic amount transiting through the wireless infrastructure and the distance between the source and the destination. Then, they propose a heuristic approach based on Hungarian Algorithm, denoted by `Hungarian-HDCN`, to solve the problem. Typically, `Hungarian-HDCN` starts by defining a utility matrix U in which each entry corresponds to the utility of the link connecting two nodes in the network. Besides, `Hungarian-HDCN` repetitively performs Hungarian algorithm on U during each iteration, in order to compute the maximum weighted matching. The Matching associates to each communication link the corresponding wireless or wired channel. At each iteration, the network utility is updated by subtracting the traffic from the new allocated links. The process is repeated until all the entries in the utility matrix

Algorithm 2: `Hungarian-HDCN` pseudo-algorithm

```
Inputs: HDCN, $m$ ongoing communications
Output: $optimal\ matching\ M$
$M \leftarrow \emptyset$
$U \leftarrow$ Compute Initial Utility Matrix
while $U \neq 0$ do
    $M \leftarrow$ Compute-MaximumWeightedMatching-Hungarian
    Set up links and allocate traffic
    $U \leftarrow$ Update Utility Matrix
```

become null, in which case all the wireless links are assigned to communications. Based on this approach, the best solution is greedily reached. Unlike the aforementioned work [8] [9], the authors assume that a wireless communication can be simultaneously transmitted through multiple links and adopt, hence, a dynamic programming approach to handle this distinction. It is worth noting that the channel allocation decision is made according to the already transmitted traffic which may affect the quality of solution in case of sporadic traffic demand. The pseudo-algorithm of `Hungarian-HDCN` is summarized through Algorithm 2.

- In [56], a new wireless link scheduling in wireless DCN is propounded. It is worth noting that the scheduling corresponds to setting up wireless links so as to alleviate congestion on hot nodes, while properly allocate channels to avoid interference. Formally, the wireless scheduling problem is modeled using two optimization objectives. The first formulation is a Min-Max optimization problem that aims to minimize the maximum remaining utility (defined in [10]) after a transmission period while satisfying interference constraint. In doing so, the authors deal with the unbalanced traffic distribution. To solve the Min-Max problem, they propose a Greedy-based algorithm, named `MM-Scheduling`. Specifically, `MM-Scheduling` repetitively selects the hottest pending node v and seeks to allocate all the transmissions through v as long as a wireless link is available. The process is repeated until all pending nodes are allocated. `MM-Scheduling` is described in Algorithm 3.
  The second formulation aims to maximize the total network utility. The authors makes use of their previous heuristic-based approach `Hungarian-HDCN` [10] to solve the best-effort optimization problem. Simulations results compare the effectiveness of the two approaches and show that `MM-Scheduling` outperforms `Hungarian-HDCN` in the case of uniform traffic distribution, while the two proposals reach similar results for hotspot traffic.

- In [62], the authors conceive a hybrid DCN architecture based on Fat-Tree design. The main idea of their proposal is to combine wired and wireless links in the same communication path. Specifically, this approach aims at minimizing hotspots formation by proposing a new logical topology for the DCN that considers IP address assignment and traffic engineering

Algorithm 3: `MM-Scheduling` pseudo-algorithm

```
Inputs: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, set of available channels $C$, set of traffic demand $T(\mathcal{E})$
Output: Channel allocation scheme $S$
$S \leftarrow 0$
$\mathcal{V}_p \leftarrow \mathcal{V}$
while $\mathcal{V}_p \neq \emptyset$ do
    $v \leftarrow$ Select-Hotest-Pending-node
    if $v$ has no available antenna OR has no remaining traffic then
        $\mathcal{V}_p \leftarrow \mathcal{V}_p - v$
    else
        $e \leftarrow$ Select a random transmission including $v$
        $c \leftarrow$ Select a random available channel on $e$
        Assign $c$ for $e$
        $S(e, c) \leftarrow 1$
return S
```

scheme. However, we notice that, they only add the wireless links in the neighborhood of the source node, while wired links are used only to deliver traffic between relay ToRs leading to the final destination.

3.3.3 Beamforming based strategies

Despite the undeniable success of 60 GHz technique and its role in enhancing wireless DCN performance, it raises the challenge of the short transmission range. In this context, we noticed that a few recent approaches have explored the use of beamforming mechanism to carry direct inter-rack communication links in HDCN. Hereafter, we will discuss the main strategies deploying directional antennas for one-hop transmissions.

- In [46], the authors explore the feasibility of the 3D beamforming primitive in data centers. Based on experimental testbed design, they prove that this technique enhances wireless links capacity and further alleviates interference compared to 2D beamforming. Moreover, this approach augments the number of current wireless transmissions in the DCN. Specifically, they show that 3D beamforming technique eliminates link blockage thanks to the ceiling reflectors. Consequently, any two racks in the DCN can communicate directly with each others using only one hop link, without the need for routing. Nevertheless, this paper only focuses on studying the feasibility of 3D beamforming technique but does not address the wireless channel allocation issue in HDCN.

- In [47], the authors extend their prior work [46] by further tackling the wireless channel as-

Algorithm 4: Greedy-HDCN pseudo-algorithm

```
Inputs: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, set of available channels $C$
Output: Channel allocation scheme $S$
$S \leftarrow 0$
$L \leftarrow$ Set of non-scheduled ongoing communications
while $L \neq \emptyset$ do
    Compute the conflict degree of each link in $L$
    Sort the set of concurrent links according to the conflict degree
    $e \leftarrow$ Link-With-Highest-Conflict-Degree
    $c \leftarrow$ Allocate-Channel($e$)
    $S(e, c) \leftarrow 1$
    $L \leftarrow L - e$
return $S$
```

signment problem in HDCN. Basically, their purpose is to address the short range and link blockage limitations of the 60 GHz technique by deploying 3D beamforming mechanism. The main contribution consists in building a small experimental testbed to prove the capacity of 3D beamforming to address the above challenges. Next, they propose a heuristic-based link scheduler algorithm, named Greedy-HDCN to allocate channels for ongoing communications. Typically, their proposal, makes use of a greedy heuristic so that the number of allocated concurrent links is maximized. To do so, the interference level of each link is estimated by computing the predictable $SINR$ (see section 4.2) values on conflicting edges. Then, the graph coloring is performed on links in such a way that conflicting edges have to be colored with different colors (i.e., channels). The main idea of the Greedy-HDCN heuristic is to sort the edges according to their conflict degrees (i.e., number of non-scheduled interfering edges). Then, channels are allocated to links in a greedy fashion. This approach is processed in a centralized manner by the centralized controller of the HDCN. We summarize the proposal through the pseudo Algorithm 4.

However, we notice that Greedy-HDCN is non-preemptive, as it keeps unchanged the channels of ongoing communications. Moreover, it requires a mechanical rotation mechanism to frequently rotate antennas inducing, hence, an extra delay. Further, this approach is very specific to the 3D beamforming based HDCN, where each two racks can directly communicate in only one single link. In fact, using only small number of racks, mirrors are used to reflect signals between racks, so that to avoid multi-hop communications. However, this can not be deployed for large scale DCNs due to the physical challenges and construction costs.

- In [63], the authors propounded a new fully wireless DCN topology arranging all racks in a single hexagonal arrangement instead of the classical row one. They made use of IEEE

802.15.3.c standard [44] to deploy the 60 GHz wireless links. Not only this approach makes possible the communication between adjacent racks, but also it enables communications between servers in the same rack, by adequately positioning the transceivers to form a polygon. Indeed, the authors enabled transceivers rotation (i.e., beam steering mechanism) in order to communicate with racks in different orientations via only point-to-point links. Note that this approach assumes that each rack has only two transceivers which limit the number of communications that can be simultaneously performed by a node. Moreover, since each node can communicate with only two neighbors simultaneously, multi-hop communications was not the prior focus of this paper. In fact, they only refer to a MAC layer mechanism [64] to deal with two-hop communications.

As a first contribution of this thesis, we will propose a new wireless channel allocation mechanism for inter-rack communications in HDCN. Our approach, denoted by `GC-HDCN`, leverages the wireless infrastructure in order to enhance network performance. Unlike [10], we assume that the DCN traffic is unsplittable and hence carried through a single channel. Besides, while the channel scheduling mechanism in [9] accords high priority to ongoing traffics, our proposal does not distinguish between incoming communications and aim to enhance the overall QoS required by applications. Contrarily to [46] [47], we assume both omni-directional and 2D directional antennas in order to avoid rotation delay induced by 3D beamforming transceivers. Moreover, we establish wireless links only between racks in the same transmission range. In doing so, we overcome the physical challenges of 3D beamforming technique that requires perfect ceiling positioning in DCNs.

3.4 Online Joint Routing and Wireless Channel Allocation strategies in HDCN

Although 60 GHz technique provides additional bandwidth to data center applications, prior proposals studied so far, restricted the wireless communications to the neighboring racks while carrying one-hop transmissions. This assumption dramatically limits the distance and the number of wireless links deployed in HDCN. Moreover, despite the ability of 3D beamforming to overcome short range limitation, it entails several physical challenges.

In this regard, a recent research approaches have dealt with multi-hop communications in HDCN. Although this issue has been heavily studied in the literature in the context of Mesh networks [65] [66] [67] [68], the related approaches ensure only fully wireless paths which is unfortunately not applicable to HDCN.

In this section, we first summarize the most relevant related work in the context of Mesh network, that helped us to have an insight into joint routing and channel assignment in Hybrid DCN.

Next, we review the main related strategies dealing with with multi-hop communications in HDCN in online mode.

3.4.1 Joint routing and channel assignment in Mesh networks

- In [69], the authors make use of multi-commodity flow model to deal with single joint routing and channel assignment in multi-channel wireless mesh networks. They aim to find the suitable routing path with the channel assignment for each communication while minimizing traffic effects. They propose a heuristic algorithm that succeed at solving the routing model in polynomial time. However, their proposal cannot be applied to a batched arrivals of requests since it accommodates only one single communication flow at once.

- In [70], the authors address the same problem but for a batch of communication flows in multi-hop wireless networks. However, their approach doesn't ensure the channel assignment along the routing paths. Indeed, the authors seek to minimize the contention effects between the ongoing links, without prohibiting it.

- In [71], the authors tackle the problem of joint routing and resource allocation in wireless data networks. They formulate the problem based on an Integer Linear Programming (ILP) statement, and make use of a dual decomposition method to solve it. This approach does not take into consideration interference constraint in the routing path. Moreover, it enables completely wireless communication routes, which is not always the case for HDCN architecture.

- A rich research work as in [72] [73], have reviewed the joint routing and channel assignment in multi-channel wireless mesh networks.

Unfortunately, these mechanisms can not be applied in the context of HDCN, where both wireless and wired interfaces have to be considered. Moreover, in HDCN, additional constraints have to be considered during the decision process. Namely, wireless interferences and the length of IP queues (waiting delay) should be jointly optimized to enhance the routing of communication flows.

3.4.2 Online joint routing and channel assignment strategies in HDCN

The joint routing and channel assignment strategies in HDCN provide the hybrid (wireless/wired) routing path for each single incoming communication request, in an online way. Hereafter, we will review the main relevant strategies found in the literature.

- In [6], the authors propound a new augmented data center architecture by deploying the 60 GHz wireless technology in their proposed `VL2` architecture. A `Greedy-Flyway-HDCN` strategy is proposed and greedily augments the wired DCN with extra flyways. The latter are

Algorithm 5: `Greedy-Flyway-HDCN` pseudo-algorithm

```
Inputs: HDCN, set of available channels $C$, set of communications $\mathcal{C}$
Output: $\mathcal{F}$ Flyway links
$\mathcal{F} \leftarrow \emptyset$
$H \leftarrow$ Set of Hotspot links
while $H \neq \emptyset$ do
    $h \leftarrow$ Select-Hotspot
    if HotSpot-On-Source then
        $f \leftarrow$ Choose-Flyway-From-Source
        Allocate-Channel-ToFlyway($f$)
    else
        /*Flyway in Destination*/
        $f \leftarrow$Choose-Flyway-To-Destination
        Allocate-Channel-To-Flyway($f$)
    $\mathcal{F} \leftarrow \mathcal{F} \cup f$
    Construct-Routing-Path($f$)
    $H \leftarrow H \setminus h$
return $\mathcal{F}$
```

60 GHz wireless links which are set up between top-of-rack switches as long as there is network congestion. In doing so, bandwidth capacity is increased. Note that each flyway is considered as i) 1-hop wireless communication and ii) not involved in the routing process. If the state of wired network is not loaded, wired infrastructure `VL2` routes the traffic using wired link-state IP routing, Open Shortest Path First (OSPF), and Equal-Cost Multi-Path (ECMP) protocols. In the case of congestion, a flyway is setup and the appropriate route is statically updated at the ToR so that the traffic passes through the wireless links. Note that each flow must transit through exactly one flyway. `Greedy-Flyway-HDCN` focuses on alleviating congestion effects by statically including flyways in wired routing paths. In other words, the proposal deals only with hotspots links. Unfortunately, the wireless channel allocation and wireless multi-hop are not considered since only non-interfering flyways are greedily added.

The pseudo-algorithm of `Greedy-Flyway-HDCN` is summarized in Algorithm 5.

- In [21], the authors propose a fully wireless data center architecture named Cayley data center topology. Racks are attached to densely wireless connected mesh topology in aim to maximize the number of active wireless links. In order to alleviate interference effects, this strategy makes use of beamforming technique with fixed-direction antennas. The routing is based on a geographic approach, denoted `XYZ-Routing`, which finds the intra and/or inter rack path. In fact, the next hop server is the closest one to the final destination. We notice that

Algorithm 6: `XYZ-Routing` pseudo-algorithm

```
Inputs: Cayley HDCN, communication $\mathcal{C}$, $g_{src}$
Output: routing path $\mathcal{P}$
$g_{curr}$ ←geographical position of the server containing current pacet
$r_{curr}$ ←rack of the current server
$g_{dst}$ ←geographical position of the final destination
$r_{dst}$ ←rack of the final destination
$\mathcal{R}_{adj} \leftarrow$ Set of racks adjacent to $r_{curr}$
$g_{curr} \leftarrow g_{src}$, $\mathcal{P} \leftarrow g_{curr}$
while $g_{curr} \neq g_{dst}$ do
    if IsInDifferentRack($g_{curr}, g_{dst}$) then
        $r_{next} \leftarrow$ Get-Min-Distance-Rack($r_{dst}, \mathcal{R}_{adj}$)
    else
        /*same rack but different servers*/
        $g_{next} \leftarrow$ Get-Min-Distance-Rack($g_{curr}, g_{dst}$)
    $\mathcal{P} \leftarrow \mathcal{P} \cup g_{curr}$
return $\mathcal{P}$
```

the authors focus only on minimizing the routing path length. Indeed, the routing decision only depends on the geographic position of the destination. In doing so, some wireless links may be excessively used and induces high probability of collisions which mitigates network performance. Moreover, this strategy does not consider wireless channel allocating jointly to the routing process. Instead, wireless channels are arbitrated based on a MAC layer arbitration protocol along the path.

The geographical routing protocol `XYZ-Routing` is summarized in Algorithm 6.

- In [49], the authors propose spherical mesh topology for wireless DCN. The racks within the same wireless transmission range are regrouped into a spherical unit. The main idea is to take profit of the geometric characteristics of the spheres to eliminate link congestion by placing antennas over them. The routing algorithm, named `Spherical-HDCN`, is based on geographical approach that gets the route depending on the position of the spheres containing the two communicating servers. Unfortunately, we notice that this strategy is very specific for the above particular spherical topology and cannot be applied to the common DCN architectures. Moreover, the proposal does not take into consideration channel assignment along the routing path.

- In [7], the authors explore the wireless infrastructure only for the control plane while data is completely transiting over wired infrastructure. The objective is to ensure a highly available control functions by alleviating interference effects and enhancing the throughput. To do so, 3D beamforming using horn/array antennas with static directions are deployed. Note that the calibration of directions aims to minimize the inter-flow interferences. In addition, new routing algorithm based on Kautz graph is proposed for signalization traffic. The key idea of this algorithm is to seek for the shortest path. Unfortunately, wireless channels over the routing path are assigned based on a simple greedy heuristic that minimizes intra-path interference but does not nullify it. Besides, the use of static 3D antennas direction strongly limits the usage of spectrum. Finally, this strategy only investigates the wireless links in the control plane, and does profit from this promising technology to alleviate massive traffic explosion in the data plane. Therefore, the proposed routing approach can not be applied to deal with inter-rack communication in modern HDCNs.

- In [41], the authors make use of free-space optical technique to augment data center network with wireless links. The wireless links are established by deploying mirrors and lens on ToRs. Note that their optical architecture ensures free-interference wireless communication links. They formulate the routing problem using a the maximum weighted matching and solve it based on a heuristic selecting minimum hop-count alternating paths. Nevertheless, this approach only considers the hop count during the routing process, since the optical technique does not require the wireless channel assignment along the path. In doing so, several important network metrics are neglected, such as the waiting delay in IP queues, link congestion, etc.

- In [74], the authors investigate, from a cross-layer view, the use of wireless infrastructure to augment the wired DCN so that to alleviate link over-subscription. This strategy separately tackles the routing and wireless channel allocation problem. In fact, first, a routing protocol is proposed to minimize the hop counts of the routing flow path. The main idea is to establish wireless links only if they reduce the total number of hops. Besides, the authors deal with congestion problem by proposing an online wireless channel and power allocation algorithm. Indeed, contrarily to most of research works dealing with HDCN, they assume that the transmission power of wireless antennas is not fixed, and propose, hence, a Greedy-based heuristic to repetitively allocate the channel ensuring the maximum capacity gain. It is worth noting that this approach may not be efficient as it computes first the shortest routing paths without considering potential channel allocation. In fact, addressing jointly the two problems is more likely to optimize the wireless resource usage. Moreover, the proposals are validated for a small instance of DCN, composed by only 20 racks, and their efficiency for large-scale DCN is not guaranteed.

As a second contribution of this thesis, we propose a new online joint routing and channel assignment approach in HDCN, for inter-rack communications, while making use of 2D beamforming technique. Unlike [49], we assume common hybrid data center network architecture based on the well known CLOS design, and our approach is not specific to a particular topology. Moreover, we do not assume static antennas' directions as in [21], so that we maximize the usage of wireless interfaces. To overcome the rotation delay induced by horn antennas in [7], we make use of 2D switched beam antennas. Unlike [41] [74], we take into account interference constraints during the routing decision. Indeed, it is not only the hop count that is considered during the path computation, but also other cost metrics. Our approach promotes the paths that ensure the higher throughput by reducing interference effects. Unlike [21], we pay attention to the link state during routing decision by prioritizing both wireless and wired interfaces with higher residual bandwidth in aim to enhance network performance. Hence, each routing communication path may be composed of wireless and/or wired links. Further, we deploy IEEE 802.11ad [23] to build 60 GHz wireless infrastructure instead of IEEE 802.15.3.c. standard, deployed in [21]. In fact, IEEE 802.11ad is better in terms of bandwidth and number of available channels. Finally, unlike [6], each routing communication path may be composed of wireless and/or wired links.

3.5 Joint Batch Routing and Channel Allocation strategies in HDCN

While the above related strategies process each single communication flow in an online way, few recent research approaches have dealt with the problem in a batch mode. The main objective of such a mode is to handle the unbalanced and heavy traffic, by carrying the batched arrivals of communication flows, and hence to ensure a better use of HDCN resources. In doing so, the communications, arriving during a specific time window, are queued together and their processing is delayed to the following time window.

Note that, there is a variety of research work addressing the joint batch routing and channel allocation in wireless mesh networks, as in [72] [73] [70]. Unfortunately, the latter mechanisms are different from our problem (HDCN), where both wireless and wired interfaces must be considered.

Hereafter, we will discuss the main few research strategies dealing with the joint batch routing and channel assignment in HDCN.

- In [50], the authors propose a `RUSH` framework for joint: i) routing and ii) scheduling wireless antennas in HDCN in both online and batch modes. They design a 3-layer multi-rooted DCN topology where each rack is equipped with only one 60 GHz steerable directional antenna. Specifically, one antenna may be involved in many routing paths simultaneously. To do so, `RUSH` allocates non-overlapping time slots for different links, while minimizing the congestion load in the HDCN. The joint routing and scheduling problem in HDCN (`JRSH`) is formulated as an Integer Linear Programming model, and has as objective to minimize the maximum link congestion. In batch mode, `RUSH` framework makes use of `RUSH-batch` algorithm. The main idea of the latter is to relax `JRSH` problem and then solve it using an LP solver. `RUSH-batch` makes use of the LP fractional solution to randomly choose routing

Algorithm 7: `RUSH pseudo-algorithm`

```
Inputs: Request set $\mathcal{R}$, the solution to the LP-relaxation of JRSH
Output: Routing scheduled paths $\mathcal{P}$
$i \leftarrow 0$
for all request $r_i$ in $\mathcal{R}$ do
  for all link $e$ transmitting flow do
    Find the single path from $s_i$ to $d_i$ through $e$ with minimum congestion load
  end for
  $p_i \leftarrow$ Pick a path
  Find a feasible scheduling on $P$
  $\mathcal{P} \leftarrow \mathcal{P} \cup p_i$
end for
```

paths for each request. Besides, based on the congestion level on each path, a feasible antenna scheduling along the path is determined. In the online mode, the authors put forward a `RUSH-online` algorithm that sequentially computes the single shortest routing path while scheduling time slots. Note that `RUSH` strategy deploys beam steering to change the antenna direction during each time fraction, which may induce extra delays. The pseudo-code of the batch algorithm of `RUSH` framework is summarized in Algorithm 7.
The same `RUSH` mechanism was used by the authors of [75], to find the hybrid routing path in the HDCN after a virtual machine deployment in the racks.

- In [52], the authors propound a new DCN architecture, `VLCcube`, by augmenting the Fat-Tree topology with optical wireless infrastructure. Specifically, all inter-rack communications are carried on only wireless links, using the visible light communication (VLC) techniques. The authors propose a new routing scheme that greedily seeks for the least congested hybrid path for each flow in both online and batch mode. Note that the proposed approach is very specific to `VLCcube` topology, since path computation depends on both the rack and pod placement. Moreover, the strategy only deals with routing problem regardless interference constraints and channel allocation problem since optical wireless communications are deployed.

- In [76], the authors deal with dual-hop routing for a set of communications requests (i.e., batch mode), in wireless dual-hop networks based on 60 GHz. Typically, they always assume a 2-hop networks where the hop count in the network can at most be equal to 2. The authors propound a decomposition heuristic method, `Dual-Heuristic`, to jointly optimize relay and link selection. The main objective of this strategy is to minimize the Maximum Expected Delivery Time. To do so, `Dual-Heuristic` decomposes, first, the original problem into a: i) relay selection, and ii) link selection sub-problems, then, it develops a Greedy heuristic to alleviate time complexity. Note, however, that is approach is very restricted to a specific configuration where 60 GHz wireless technique is used only for two hops, and can not be applied in the context of HDCN. Moreover, it does not deal with channel assigning alongside the routing process.

The third contribution of this thesis consists in proposing a new joint batch routing and channel assignment approach in HDCN, to deal with the batched arrivals of communication flows. It is worth pointing out that none of the previous strategies address the channel allocation jointly to the routing process in batch mode. Contrarily to [50], our approach deals with a batch of flows while allocating wireless channels along the paths. Moreover, unlike [52], we design a hybrid DCN by augmenting the wired network with wireless communication links, and our proposal is generic and is not specific to a particular HDCN topology. Finally, contrary to [76], our proposed algorithm does not limit the number of wireless links in the hybrid routing path.

3.6 Summary

Table 3.1 summarizes a comparison between the aforementioned strategies for: i) wireless channel allocation and ii) online and batch joint routing and channel assignment, in HDCN. Specifically, we classify the related method according to the: i) deployed architecture, ii) addressed problem (i.e., one-hop or multi-hop communications), iii) processing mode (i.e., online or batch), iv) constraints considering during the decision, and v) deployed technique.

3.7 Conclusion

In this chapter, we provided a detailed overview of routing and channel allocation strategies in HDCN, for both one-hop and multi-hop inter-rack communications. First, we briefly described the wireless channel allocation problem for intra-DCN flows in single hop, and the joint routing and wireless channel assignment problem for multi-hop communications. Then, we addressed the main challenges encountered by this issue in HDCN. Afterwards, we highlighted the most important criteria that have been considered when dealing with the routing and wireless channel allocation problems in HDCN. Next, we detailed the main related strategies that we classify into three main groups: i) wireless channel allocation approaches dealing with one-hop communications in HDCN, ii) online joint routing and wireless channel allocation approaches addressing multi-hop communications in HDCN in a sequential way, and iii) batch joint routing and wireless channel assignment approaches handling the batched arrivals of communication flows to HDCN. Finally, we summarized the review with a qualitative comparison of the different proposed strategies.

In this thesis, we address the challenges of routing and wireless resource allocation in HDCN by tackling the problem in three stages. In each stage, we propose a new strategy having the same focus of each group of the above taxonomy. In the next chapter, we will present our first contribution dealing with wireless channel allocation in HDCN. The proposal will focus only on single-hop inter-rack communications.

Table 3.1: Summary of routing and channel allocation strategies in HDCN

Strategy	Architecture	Solution	One/Multi hop	mode	Constraints	HDCN Technique	wireless technique
Genetic-HDCN [8]	Tree-layered	heuristic	one-hop	online	interference	wireless/wired	omni-directional 60 GHz
Approximation-HDCN [9]	Tree-layered	approximative	one-hop	online	interference	wireless/wired	omni-directional 60 GHz
Hungarian-HDCN [10]	Tree-layered	heuristic	one-hop	online	interference	wireless/wired	omni-directional 60 GHz
MM-Scheduling [56]	Tree-layered	heuristic	one-hop	online	interference	wireless/wired	omni-directional 60 GHz
[62]	Fat-Tree	heuristic	one-hop	online	interference	wireless/wired	omni-directional
[46]	tree-based	-	one-hop	online	interference	wireless/wired	3D beamforming
Greedy-HDCN [47]	tree-based	heuristic	one-hop	online	interference	wireless/wired	3D beamforming
[63]	hexagonal Fat-Tree	-	one-hop	online	interference	fully wireless	beamforming 60 GHz
Greedy-Flyway-HDCN [6]	VL2	Greedy	routing	online	interference	wireless/wired	beamforming 60 GHz
XYZ-Routing [21]	Cayley DCN	geographic	routing	online	path length	wireless	beamforming 60 GHz
Spherical-HDCN [49]	spherical DCN	geographic	routing	online	distance	wireless	beamforming 60 GHz
[7]	Angora	Kautz-graph	routing	online	hop number	wireless/wired	3D beamforming
[41]	FireFly	heuristic	routing	online	hop-count	wireless optics	Free-space optics
[74]	grid	decomposition method	routing	online	power	wireless	60 GHz
RUSH [50]	3-layered	ILP relaxation	routing	online/batch	time scheduling	wireless/wired	beamforming 60 GHz
VLCcube [52]	Fat-Tree	heuristic	routing	batch	rack placement	wireless optics	VLC
Dual-Heuristic [76]	Dual-hop network	heuristic	2-hop routing	batch	delivery time	wireless/wired	beamforming 60 GHz

Chapter 4

Wireless channel allocation for one-hop communications in HDCN

Contents

4.1 Introduction

In this chapter, we will address the issue of wireless channel allocation in hybrid data center networks. The main objective is to efficiently allocate wireless channels for single-hop intra-data center communications in such a way that enhances the HDCN throughput and minmizes congestion effects. It is undeniable that deploying the wireless 60 GHz technique in HDCN has several advantages. However, such an architecture is faced with two significant challenges. Firstly, the number of wireless channels available in the physical layer and their bandwidth capacities are limited. Secondly, a wireless channel cannot be assigned to more than one wireless communication at the same time in the interference area. Otherwise, collisions will occur in the medium and consequently the QoS will be deteriorated. To get rid of the aforementioned challenges, we have, first, designed a Hybrid DCN architecture making use of Cisco's Massively Scalable Data Center (MSDC) model [22], detailed in Section 2.4, based on both i) IEEE 802.11ad (wireless) and ii) Ethernet (wired) standards. Then, we propose a new wireless resource allocation algorithm, named resource allocation algorithm based on **Graph Coloring in Hybrid Data Center Network** (`GC-HDCN`). The objective of `GC-HDCN` is to maximize the total throughput supported in the DCN. The main idea of our approach is to maximize the proportion of one-hop intra-data center communication requests transiting over the wireless infrastructure and the rest will be transmitted over the wired infrastructure. In doing so, the end-to-end delay of communications and the congestion of wired infrastructure are minimized.

The remainder of this chapter is organized as follows. In Section 4.2, the wireless resource allocation problem within HDCN will be formulated. Afterwards, Section 4.3 will describe the details of our proposal `GC-HDCN`. Simulation environment and performance evaluation will be presented in Section 4.4. Finally, Section 4.5 will conclude the chapter.

4.2 Problem Formulation

In this section, we will formulate the wireless channel allocation problem in HDCN. We will first describe the model of inter-rack wireless communications. Then, we will detail the problem formalization based on a Minimum Graph Coloring approach.

4.2.1 Hybrid Data Center Network Model

Each Wireless Transmission Unit (WTU) denoted by $\mathcal{W}_i$, is equipped with 4 IEEE 802.11ad transceivers/antennas denoted by $\{w_i^1, w_i^2, w_i^3, w_i^4\}$. The number of antennas depends on the number of orthogonal channels available in IEEE 802.11ad standard. We recall that each $\mathcal{W}_i$ is deployed over the top of the rack and the wired infrastructure coexists with the wireless transmission units. The communications over the racks are ensured by the $\{\mathcal{W}_i\}$ and/or the gigabit wired (ToR) switches. Our objective is to maximize the number of communications transiting over the wireless infrastructure in order to minimize the congestion of the wired infrastructure.

We model the set $\mathcal{C}$ encompassing the ongoing wireless communications (i.e., accepted in the Hybrid DCN) and the new incoming communication request (i.e., $\mathcal{C} = \{c_i^j\}$), as an undirected graph $\mathcal{G}$= ($\mathcal{V}$, $\mathcal{E}$). Each node $n \in \mathcal{V}$ corresponds to one communication c_i^j from the transmitter $\mathcal{W}_i$ to the receiver $\mathcal{W}_j$. Obviously, for each communication $c_i^j = (\mathcal{W}_i, \mathcal{W}_j)$, $\mathcal{W}_j$ is located within the IEEE 802.11ad transmission range T_R of $\mathcal{W}_i$. An edge $e = (c_i^j, c_k^l) \in \mathcal{E}$ exists only if c_i^j is susceptible to interfere with c_k^l or vice versa. We model the interference between two wireless communications c_i^j and c_k^l as follows: i) transmitter $\mathcal{W}_i$ interferes with receiver $\mathcal{W}_l$ or ii) transmitter $\mathcal{W}_k$ interferes with receiver $\mathcal{W}_j$.

We make use of the Friis signal transmission model. In fact, we assume that obstacles do not exist in the data center environment and radio antennas are deployed on the top of racks. The receiving signal power sent by w_i^k to w_j^k is equal to:

$$P_r(i,j,k) = P_t + G(\theta(i,j,k)) + 20\log_{10}\left(\frac{\eta}{4\pi d}\right)^{\alpha} - \tau - \psi \tag{4.2.1}$$

where i) P_t is transmitting signal power, ii) $G(\theta(i,j,k))$ is the gain of transmitting and receiving antennas and $\theta(i,j,k)$ refers to the azimuth angle between antennas, iii) η (meter) is the wavelength, iv) d (meter) is separating distance between w_i^k and w_j^k, v) α is the path loss effects, and vi) τ and ψ are respectively the noise factor and the implementation loss fixed in IEEE 802.11ad standard. In this thesis, similarly to [8], we adopt the interference disk model. It is worth noting that such a model is independent of the antenna technique. It relies only on the physical position of transmitting nodes and the active channels. The Signal to Interference Noise Ratio between transmitter w_i^k and destination w_j^k on the channel k is equal to:

$$SINR(i,j,k) = \frac{P_r(i,j,k)}{\sum_{m \neq i} P_i(m,j,k)} \tag{4.2.2}$$

$P_i(m,j,k)$ is the interference power received at antenna w_j^k and caused by w_m^k on the beam used in the communication initiated by w_i^k. It is worth noting that w_i^k succeeds to communicate with w_j^k if and only if $SINR(i,j,k)$ and $SINR(j,i,k)$ (i.e., ACK frames reception) are at least equal to CP_Thr. The latter is a hardware constant of the transceiver. Accordingly, two communication $c_i^j = (w_i^k, w_j^k)$ and $c_m^n = (w_m^k, w_n^k)$ interfere on channel k if: i) transmitter, w_i^k of c_i^j interferes with receiver w_n^k of c_m^n or ii) transmitter w_m^k of c_m^n interferes with receiver w_j^k of c_i^j.

Given the static topology of racks in the HDCN, we initially compute the $SINR$ table containing all the signal-to-noise ratio values between all the racks for different antennas orientations (i.e., beams). It is worth noting that entries in this table are opportunistically refreshed, during the ongoing wireless traffic transmissions. In fact, we measure signal strength received from active sending racks at different antenna orientations. Then, signal measurements and transmitter antennas' orientations are shared by the CC node, using the wired infrastructure. Note that thanks to $SINR$ table, it is possible not only to compute interference at each beam, but also to determine the best antenna orientation for two communicating ToRs. Moreover, since in our architecture both data and Ack packets are transmitted over wireless infrastructure, therefore, for each communicating ToR, two (for sending and reception) beams are identified. By the incoming of each new communication, the communicating antennas are configured to the suitable beam that ensure the best gain. Then, the entries of $SINR$ table are refreshed while taking into consideration the new and ongoing antenna orientations of all the racks in the HDCN. The interference/communication graph $\mathcal{G}$ is, hence, re-constructed.

4.2.2 Wireless Channel allocation problem in HDCN

Our objective is to maximize the proportion of c_i^j transiting over the wireless IEEE 802.11ad network in order to minimize the congestion level of wired infrastructure. To do so, the wireless channel allocation must be optimized. In fact, the decision is made at the arrival of each new communication request. Consequently, the switching of wireless channels (i.e., hop channel) at the physical layer for the wireless communications is permitted. In other words, a wireless communication can modify its physical channel and continues its transmission over the new assigned one. Furthermore, a current wireless communication can switch to the wired infrastructure.

One important issue that must be taken into consideration for channel allocation is the interference between wireless communications. To achieve our goal, channels are dynamically assigned to the communications while taking into account the potential interference between them. Moreover, due to the limited number of channels (i.e., 4), unassigned communications are carried through the wired links.

We formulate the wireless channel allocation problem as a Minimum Graph Coloring Problem (Min-GCP) [77] in such a way that each node $n \in \mathcal{V}$ (i.e., communication) will have exactly one color, while guaranteeing that two adjacent nodes have different colors. Note that, henceforth, the objective is to minimize the number of colors used to cover all graph nodes.

Let $\tilde{\mathcal{S}}$ denote the set of all maximal stable sets in $\mathcal{G}$. We recall that a stable set is a subset of $\mathcal{V}$ which is composed of pairwise non-adjacent nodes. A maximal stable set is a stable set that is not strictly included in any other stable set. It is worth pointing out that all the nodes in the stable set can be assigned one color since they are not neighbors. In doing so, the group of wireless communications corresponding to the nodes in the stable set make use of the same wireless channel. Our objective is to calculate the minimum number of stable sets, k covering all nodes in $\mathcal{G}$. Such a number corresponds to k-coloring and is called the chromatic number of $\mathcal{G}$. It is denoted by $\chi(\mathcal{G})$. To calculate $\chi(\mathcal{G})$, we formulate our problem as an Integer Programming (IP) based on the independent set formulation.

$$
\begin{array}{ll}
\chi(G) \texttt{ = min} & \sum_{\hat{S}\in\tilde{\mathcal{S}}} x_{\hat{S}} \\
\texttt{subject to:} & \forall n \in \mathcal{V},\ \sum_{\hat{S}\in\tilde{\mathcal{S}}} \left(x_{\hat{S}}.1_{\{n\in\hat{S}\}} \right) \geq 1 \\
 & \forall \hat{S} \in \tilde{\mathcal{S}},\ x_{\hat{S}} \in \{0,1\}
\end{array}
$$

Problem 1: Min-GCP – Wireless channel allocation problem where i) $x_{\hat{S}}$ is a binary variable defining whether $\hat{S} \in \tilde{\mathcal{S}}$ is assigned a color or not and ii) $1_{\{n \in \hat{S}\}}$ is the indication function, it is equal to 1 if the condition $n \in \hat{S}$ is true otherwise it is 0. According to the constraint, each node $n \in \mathcal{V}$ (i.e., communication) must have at least one color (i.e., wireless channel). Hence, the idea is to select only one color among those assigned to n.

It is obvious to see that the number of variables can be tremendous since it depends on the size of the graph. In fact, the Integer Programming is NP-complete [78]. Hence, making use of computational methods would not be an interesting idea since the scalability is not guaranteed. Consequently, an effective approach should be proposed to cleverly tackle the problem and efficiently converge to the best (i.e., minimum number of colors) solution.

4.3 Proposal: `GC-HDCN`

As explained above, solving the minimum coloring problem using computational methods is not reasonable due to the high number of variables. To tackle the aforementioned problem, the solution is to first address a subset of variables then progressively generate new variables when needed. This is the key idea of column generation optimization approach [77].

In this section, we will detail our proposal strategy named Graph Coloring in Hybrid Data Center Network (`GC-HDCN`) based on the column generation optimization approach. The main objective is to converge to the best solution of the minimum coloring problem. The rational behind `GC-HDCN` is to generate the maximum-sized stable sets. Each stable set is composed of a group of wireless communications that use the same wireless channel (i.e., same color).

`GC-HDCN` proceeds as following. First, Problem 1 (i.e., Min-GCP – Wireless channel allocation problem) is **relaxed** (i.e., $0 \leq x_{\hat{S}} \leq 1$) and then **resolved** while assuming an **initial subset** of maximum stable sets $\mathcal{S}_r$ generated by a **Greedy Heuristic** (`GH`) method. The relaxed problem is named **Restricted Master Problem** (`RM-Problem`) since it considers only a **subset** of maximum stable sets. Secondly, the above `RM-Problem` is solved based on an **exact** method (i.e., simplex). Note that the optimal dual variables corresponding to the constraints of `Min-GCP` are used to define a new sub problem called **Pricing problem**. The latter is solved in order to determine whether it would be useful to add a new variable (i.e., stable set) to $\mathcal{S}_r$. If the solution of the **Pricing problem** corresponds to an improving stable set, then the latter is added to $\mathcal{S}_r$ and the `RM-Problem` will be resolved again. The process will be repeated until no new improving columns (i.e., variables) can be generated and added. If the final solution of `RM-Problem` is integer then it corresponds to the optimal solution of `Min-GCP`. Otherwise, a **Branch and Price** algorithm is carried out to enforce integrality and thus find the best integer solution. `GC-HDCN` is summarized in Figure 4.1. In the rest of this section, we will detail each stage of `GC-HDCN` .

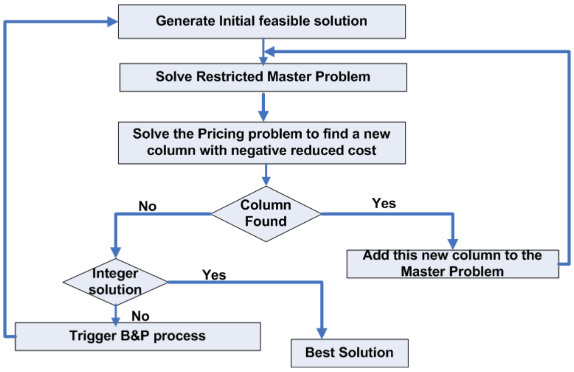

Generate Initial feasible solution
Solve Restricted Master Problem
Solve the Pricing problem to find a new column with negative reduced cost
Column Found
No
Yes
Add this new column to the Master Problem
Integer solution
Yes
No
Trigger B&P process
Best Solution

Figure 4.1: Flowchart of GC-HDCN

4.3.1 Generation of initial solution

This stage consists in generating an initial subset $\mathcal{S}_r$ of maximal independent sets. $\mathcal{S}_r$ is built using the Greedy Heuristic GH [77]. The key idea of GH is to sort the nodes $n \in \mathcal{V}$ in descending order according to their connectivity degree. Then, the highest weighted node in $\mathcal{V}$ is selected as an initial element of the first maximal independent set $\hat{S}_0$. Afterwards, remaining nodes are sequentially added to $\hat{S}_0$ while checking that the resulting set is still independent. Once $\hat{S}_0$ is built, it is added to $\mathcal{S}_r$. The process is recursively repeated to create the rest of maximal independent sets $\{\hat{S}_i\}$. Note that $\mathcal{S}_r = \cup_i \{\hat{S}_i\}$ and a node $n \in \mathcal{V}$ may belong to several independent sets $\hat{S}_i$.

4.3.2 Resolution of the relaxed RM-Problem

Once $\mathcal{S}_r$ is generated, the latter is used as an input of relaxed RM-Problem. It is worth noting that the number of variables (i.e., columns) corresponds to the size of $\mathcal{S}_r$. On the other hand, the number of constraints is equal to $|\mathcal{V}|$ (i.e., set of communications in $\mathcal{G}$). Hereafter, the definition of the relaxed RM-Problem:

$$\begin{array}{ll}
\texttt{min} & \sum_{\hat{S} \in \mathcal{S}_r} x_{\hat{S}} \\
\texttt{subject to:} & \forall n \in \mathcal{V}, \ \sum_{\hat{S} \in \mathcal{S}_r} \left(x_{\hat{S}} \cdot 1_{\{n \in \hat{S}\}} \right) \geq 1 \\
 & \forall \hat{S} \in \mathcal{S}_r, \ \ x_{\hat{S}} \geq 0
\end{array}$$

Problem 2: Relaxed Restricted Master Problem

The aforementioned relaxed RM-Problem is resolved using Simplex algorithm [79].

4.3.3 Resolution of the pricing problem

The main objective of this stage is to gradually enrich the set of maximal stable sets $\mathcal{S}_r$. The idea is to judiciously generate and add new columns (i.e., maximal stable sets) in order to converge to the optimal solution. At this stage, we determine whether it is interesting to expand $\mathcal{S}_r$ by adding new improving stable sets or not. To do so, we search in an iterative manner for the stable sets having *negative* reduced costs. Note that the reduced cost of a stable set $\hat{\mathcal{S}}$ is defined as:

$$\mathcal{R}(\hat{\mathcal{S}}) = 1 - \Pi(\hat{\mathcal{S}}) = 1 - \sum_{n_i \in \hat{\mathcal{S}}} \pi_i \tag{4.3.3}$$

where the coefficients π_i correspond to the optimal dual variables of the relaxed `RM-Problem` constraints calculated by Simplex algorithm in the previous stage (Section 4.3.2). It is straightforward to see that generating a new stable set $\hat{\mathcal{S}}$ with a negative cost is equivalent to the resolution of pricing problem with an obtained objective function greater than 1. Otherwise, we can conclude that there exist no improving independent sets. Consequently, solving relaxed `RM-Problem` over the current $\mathcal{S}_r$ is equivalent to solving Min-GCP over $\tilde{\mathcal{S}}$.

$$\begin{array}{ll} \texttt{max} & \sum_{n \in \mathcal{V}} \pi_n \cdot y_n \\ \texttt{subject to:} & \forall (n,m) \in \mathcal{E},\ \ y_n + y_m \leq 1 \\ & \forall n \in \mathcal{V},\ \ y_n \in \{0,1\} \end{array}$$

Problem 3: Pricing problem: `Pr-ILP`

It is straightforward to see that the above pricing problem, `Pr-ILP`, is an Integer Linear Programming (ILP) problem, since the variable y_n is integer and the objective function is linear.

This resolution of `Pr-ILP` aims to find the maximal stable that might improve the relaxed `RM-Problem`. It is worth noting that in today's large-scaled data centers, traffic is very heavy, and hence many ongoing communications are likely to be carried simultaneously. Consequently, the wireless transmission/interference graph $\mathcal{G}$ scales up with the traffic density. In such a case, researches claim that generating new potential stable sets based on the exact resolution of the aforementioned pricing problem, requires high computation time. To get rid of this complexity challenge, we propose, in this work, a combined heuristic/exact approach, denoted by `GH-GC-HDCN`. The key insight of our approach is to keep generating new optimal stable sets as far as the number of ongoing communications in the HDCN is less or equal to a specific threshold value Thr_D. Actually, in such a case, the graph $\mathcal{G}$ is still small-sized, and hence optimal stable sets can be computed, in a reasonable time, by resolving the pricing problem, based on Branch-and-Cut (`B&C`) algorithm. Otherwise, when the number of ongoing communication flows is greater than Thr_D, the transmission/interference graph becomes dense. Therefore, our approach makes use of the greedy heuristic, `GH+` in order to generate new columns. In fact, finding many feasible maximal stable sets with negative cost is sufficient. Since `GH+` is simple and fast, it is carried out recursively to generate the new columns. Hereafter, we will detail both `GH+` and `B&C` algorithms.

4.3.3.1 B&C algorithm

To solve the ILP formulation of `Pr-ILP` problem for small instances of the graph $\mathcal{G}$, our approach makes use of B&C algorithm.

To do so, B&C basically relies on two main techniques: i) Cutting planes and ii) Branch-and-Bound, to reach efficiently the optimal solution. First, the algorithm relaxes the ILP problem by transforming all the integer variables $y^n, n \in \mathcal{V}$ into continuous ones. Second, the relaxed linear problem (R-LP) is solved based on the regular Simplex algorithm. When an optimal solution is obtained, then, the algorithm checks whether some variables have fractional values. If such variables exist, then the algorithm cuts away parts of the solution set by adding a new linear constraint which is satisfied by all integer variables but violated by the fractional ones. Afterwards, the relaxed problem is resolved again in order to eliminate the fractional solutions while keeping the integer ones. Note that the process is repeatedly executed to improve the problem relaxation and hence become closer to the integer solution. The algorithm stops when no cutting plane can be found, or a fully integer solution is obtained.

If no additional cutting planes can be found, and the obtained solution is not integer, then B&C resorts to Branch-and-Bound (B&B). The main task of the latter consists in searching for the cutting planes in an efficient way in order to rapidly reach the optimal solution. To do so, it proceeds by partitioning the problem into new restricted regions. Then, it constructs a tree enumerating all the possible variable settings. Only some specific branches of the tree, that are expected to produce optimal/close to optimal values, are explored. The new linear problems are hence solved with Simplex algorithm and the process is repeated.

4.3.3.2 Pricing Greedy heuristic

We make use of a variant of GH defined in section 4.3.1 denoted by GH+. In fact, the new Greedy heuristic generates at most N_{max} promising (i.e., negative reduced cost) maximal independent sets (i.e., column) and the weight function w_n of each node n takes into account the calculated cost in the relaxed RM-Problem. Formally, the weight of a node n is defined as:

$$w_n = \sqrt{{c_n}^2 \cdot \pi_n^2} \tag{4.3.4}$$

where c_n is the connectivity degree of n in $\mathcal{G}$ and π_n denotes its dual value. To do so, GH+ sorts the nodes $n \in \mathcal{V}$ in a descending order according to their weights. Then, the highest weighted node in $\mathcal{V}$ is selected as an initial element of the maximal independent set $\hat{S}_i$, $i \in \{1, ..., N_{max}\}$. Afterwards, remaining nodes are sequentially added to $\hat{S}_i$ as long as the resulting set is still independent. Once $\hat{S}_i$ is built, it is added to $\mathcal{S}_r$, i.e., $\mathcal{S}_r = \mathcal{S}_r \cup \{\hat{S}_i\}$. Note that GH+ is recursively repeated to create the rest of maximal independent sets $\{\hat{S}_i\}$. The process stops if N_{max} maximal independent sets have been generated or when no new column can be found. Similarly to the initial solution, a node $n \in \mathcal{V}$ may belong to several independent sets $\hat{S}_i$.

The pseudo-algorithm of GH+ is summarized in Algorithm 8. Afterwards, selected stable sets are used as the input of relaxed RM-Problem.

Algorithm 8: Pricing stage: `GH+`

```
Inputs: $\mathcal{G}$, $\{\pi_n\}$, $N_{max}$
Output: $\mathcal{S}_{sel} \leftarrow \cup_i\{\hat{\mathcal{S}}_i\}$
for $n \in \mathcal{V}$ do
    $c_n \leftarrow$ Connectivity degree of $n$
    $w_n \leftarrow \sqrt{{c_n}^2 \cdot \pi_n^2}$
$\mathcal{Q} \leftarrow$ Descending sort of nodes $n \in \mathcal{V}$ w.r.t weights $w_n$
$\mathcal{S}_{sel} \leftarrow \emptyset$
$i \leftarrow 1$
$Stop \leftarrow false$
while $Stop = false$ do
    $\hat{\mathcal{S}}_i \leftarrow \emptyset$
    $n \leftarrow$ Head($\mathcal{Q}$)
    $\hat{\mathcal{S}}_{tmp} \leftarrow \{n\}$
    $\mathcal{Q} \leftarrow \mathcal{Q} \backslash \{n\}$
    for $m \in \mathcal{Q}$ do
        if disjoint $(m, \hat{\mathcal{S}}_{tmp})$ then
            $\hat{\mathcal{S}}_{tmp} \leftarrow \hat{\mathcal{S}}_{tmp} \cup \{m\}$
    if $\hat{\mathcal{S}}_{tmp} \neq \emptyset$ then
        Calculate $\mathcal{R}(\hat{\mathcal{S}}_{tmp})$
        if $\mathcal{R}(\hat{\mathcal{S}}_{tmp}) < 0$ then
            $\hat{\mathcal{S}}_i \leftarrow \hat{\mathcal{S}}_{tmp}$
            $\mathcal{S}_{sel} \leftarrow \mathcal{S}_{sel} \cup \hat{\mathcal{S}}_i$
            $i \leftarrow i + 1$
            if $i > N_{max}$ then
                $Stop \leftarrow true$
        if $(\mathcal{Q} = \emptyset)$ then
            $Stop \leftarrow true$
    else
        $Stop \leftarrow true$
```

The column generation process is recursively carried out until no new column with negative reduced cost can be generated. The column generation process is summarized in Algorithm 9, which combines the resolution of relaxed `RM-Problem` and pricing problem. Once the process converges (i.e., no more improving column), if the resulting solution of the relaxed `RM-Problem` is integer (i.e., $\forall x_{\hat{\mathcal{S}}},\ x_{\hat{\mathcal{S}}} \in \{0, 1\}$), then we can conclude that the solution is optimal [77]. Otherwise, we need to enforce the integrality. To do so, Branch-and-Price algorithm is performed to

Algorithm 9: Column generation process

```
Inputs: $\mathcal{G}$, $\mathcal{S}_r$ , $Thr_D$
Output: $\mathcal{S}_{out}$, $\{x_{\hat{\mathcal{S}}}\}$, $\hat{\mathcal{S}} \in \mathcal{S}_{out}$, $x_{\hat{\mathcal{S}}} \in [0,1]$
$k \leftarrow 0$
$Stop \leftarrow false$
while $Stop = false$ do
    $\hat{\mathcal{S}} \in \mathcal{S}_k, n \in \mathcal{V}$
    $\{x_{\hat{\mathcal{S}}}\}, \{\pi_n\} \leftarrow$ Solve relaxed-RM-Problem($\mathcal{G}, \mathcal{S}_k$)
    if ($Size(\mathcal{G}) \geq Thr_D$) then
        $\hat{\mathcal{S}} \leftarrow$ GH+ ($\mathcal{G}$, $\{\pi_n\}$, $N_{max}$)
    else
        $\hat{\mathcal{S}}_{op} \leftarrow$ Exact-Pricing-B&C ($\mathcal{G}$, $\{\pi_n\}$)
        $\hat{\mathcal{S}} \leftarrow \hat{\mathcal{S}}_{op}$
    if ($\hat{\mathcal{S}} = \emptyset$) then
        $Stop \leftarrow true$
        $\mathcal{S}_{out} \leftarrow \mathcal{S}_k$
    else
        $\mathcal{S}_{k+1} \leftarrow \mathcal{S}_k \cup \hat{\mathcal{S}}$
        $k \leftarrow k + 1$
```

compute the integer solution.

4.3.4 Branch and price stage

The main task of this stage is to enforce the integrality of variables $x_{\hat{\mathcal{S}}}$. To do so, Branch and Price (B&P) [80] is performed. B&P is a combination of B&B and column generation [77] methods. This method has good performances when the lower bound is tight which is the case of our problem [80]. B&P is carried out only if no new columns (i.e., stable sets) can be added and the solution of the relaxed RM-Problem is not integer. Branching rules are defined such as they ensure that i) the sub-problem tackled at each node in the solution tree is itself a graph coloring problem solved by column generation method and ii) the integer optimal solution is exactly supported by one branch in the solution tree.

B&P proceeds as following. First, two overlapping stable sets $\mathcal{S}_1$ and $\mathcal{S}_2$ are considered. $\mathcal{S}_1$ is selected such as is typified by the highest fractional value of $x_{\mathcal{S}_1}$. The highest fractional value corresponds to the value $x_{\mathcal{S}_1} - \lfloor x_{\mathcal{S}_1} \rfloor$ which is close to $\frac{1}{2}$. $\mathcal{S}_2$ is randomly selected such as $\mathcal{S}_1 \cap \mathcal{S}_2 \neq \emptyset$. Two nodes n_1 and n_2 are then randomly selected such as: $n_1 \in \mathcal{S}_1 \cap \mathcal{S}_2$ and $n_2 \in (\mathcal{S}_1 \setminus \mathcal{S}_2) \cup (\mathcal{S}_2 \setminus \mathcal{S}_1)$. As in [77], we define the following new coloring graph subproblems:

- $G_{same}(\mathcal{G}, n_1, n_2)$: merge n_1 and n_2 in graph $\mathcal{G}$ into a new node n^*. All edges from/to n_1 and

n_2 in $\mathcal{G}$ will be connected to n^*. The new graph generated is denoted by $\mathcal{G}_{same}$.

- $G_{diff}(\mathcal{G}, n_1, n_2)$: add a new link between n_1 and n_2 in graph $\mathcal{G}$. The new graph generated is denoted by $\mathcal{G}_{diff}$.

It is clear to see that thanks to the above branching, resulting subproblems do not define any additional constraints compared to the master problem (Problem 2). The two sub-problems (i.e., $\mathcal{G}_{diff}$ and $\mathcal{G}_{same}$) are added to the tree of branch and price in which the root is an abstract node. The resolution of coloring subproblems G_{same} and G_{diff} may add new columns (i.e., maximal stable sets) to tighten the relaxation of the relaxed `RM-Problem` and hence enforce integrality. Thanks to column generation (see Algorithm 9), we resolve the coloring problem of $\mathcal{G}_{diff}$. If the solution is integer then the process is converged. Otherwise, we resolve the coloring problem of $\mathcal{G}_{same}$. Like in the previous step, the convergence is reached if the solution is integer. Otherwise two other graphs, denoted $\hat{\mathcal{G}}_{diff}$ and $\hat{\mathcal{G}}_{same}$, are generated from the graph $\mathcal{G}_{diff}$ or $\mathcal{G}_{same}$ and added to the tree of branch and price algorithm. Thanks to Depth First Search algorithm, the leaf node (i.e., subproblem) in the `B&P` tree which is characterized by the lowest value of the objective function (i.e., $\sum(x_{\hat{\mathcal{S}}})$) is elected. The same process is recursively repeated to the elected node until integrality is reached. It is worth noting that the convergence is ensured thanks to the Branch and Bound algorithm. `B&P` stage is summarized in Algorithm 10.

4.4 Performance evaluation

In this section, we will gauge the performance of our proposed algorithms `GH-GC-HDCN` and `GC-HDCN` based on extensive simulations. First of all, we describe the three stages of our implementation, namely i) IEEE 802.11ad standard integration in network simulator QualNet[1] ii) deployment of MSDC data center architecture and iii) development of our proposed decision algorithm `GC-HDCN` and simulation environment set up. Then, we define the performance metrics to assess our proposal and the related strategies. Finally, we discuss the effectiveness of our proposal by comparing it with the most prominent related strategies, which we implemented, found in the literature: i) `Genetic-HDCN` [8] [9], ii) `Hungarian-HDCN` [10] and iii) `Wired-DCN`. Note that the latter strategy leverages only the Ethernet-based infrastructure.

4.4.1 Simulation Environment and Methodologies

4.4.1.1 Experiment Design

In order to evaluate the effectiveness of our approach `GC-HDCN` and prove its soundness in Hybrid DCNs, we proceed as follows. First, we implemented the IEEE 802.11ad standard in QualNet.

[1] http://www.scalablenetworks.com/products/Qualnet/

Algorithm 10: Branch and price

```
Inputs: $\mathcal{G}$, $\mathcal{S}_{out}$, $\{x_{\hat{\mathcal{S}}}\}$, $\hat{\mathcal{S}} \in \mathcal{S}_{out}$, $x_{\hat{\mathcal{S}}} \in [0, 1]$
Output: $\mathcal{S}_{fin}$, $\{x_{\hat{\mathcal{S}}}\}$, $\hat{\mathcal{S}} \in \mathcal{S}_{fin}$, $x_{\hat{\mathcal{S}}} \in \{0, 1\}$
$Stop \leftarrow false$
$\mathcal{S}_{tmp} \leftarrow \mathcal{S}_{out}$
$\mathcal{T} \leftarrow$ abstract root node
while $Stop = false$ do
    Select $\mathcal{S}_1 \in \mathcal{S}_{tmp} : \left|x_{\mathcal{S}_1} - \lfloor x_{\mathcal{S}_1} \rfloor - \frac{1}{2}\right| = \min_{\mathcal{S}_i \in \mathcal{S}_{tmp}}(\left|x_{\mathcal{S}_i} - \lfloor x_{\mathcal{S}_i} \rfloor - \frac{1}{2}\right|)$
    Select randomly $\mathcal{S}_2 \in \mathcal{S}_{tmp} : \mathcal{S}_1 \cap \mathcal{S}_2 \neq \emptyset$
    Select randomly $n_1 \in \mathcal{S}_1 \cap \mathcal{S}_2$
    Select randomly $n_2 \in (\mathcal{S}_1 \setminus \mathcal{S}_2) \cup (\mathcal{S}_2 \setminus \mathcal{S}_1)$
    $G_{same} \leftarrow$ Build-Same $(\mathcal{G}, n_1, n_2)$
    Column-Generation $(G_{same}, \mathcal{S}_{tmp}, \mathcal{S}^1_{out}, \{x_{\hat{\mathcal{S}}}\})$
    if $\{x_{\hat{\mathcal{S}}}\}$ are integer then
        $Stop \leftarrow true$
        $\mathcal{S}_{fin} \leftarrow \mathcal{S}^1_{out}$
    else
        $G_{diff} \leftarrow$ Build-Diff $(\mathcal{G}, n_1, n_2)$
        Column-Generation $(G_{diff}, \mathcal{S}_{tmp}, \mathcal{S}^2_{out}, \{x_{\hat{\mathcal{S}}}\})$
        if $\{x_{\hat{\mathcal{S}}}\}$ are integer then
            $Stop \leftarrow true$
            $\mathcal{S}_{fin} \leftarrow \mathcal{S}^2_{out}$
        else
            $\mathcal{T} \leftarrow$ Add-Sub-Problem $(G_{same})$
            $\mathcal{T} \leftarrow$ Add-Sub-Problem $(G_{diff})$
            Node-Tree $\leftarrow$ Leaf-Depth-First-Search ($\mathcal{T}$, “minimal", $\sum(x_{\hat{\mathcal{S}}})$)
            $\mathcal{S}_{tmp} \leftarrow$ Father(Node-Tree, $\mathcal{S}_{out}$)
```

Note that QualNet is an event driven industrial network simulator based on C++ language. It is widely used by the network research community. Its modularity and layer based architecture ease the design and the development of new protocols in whether wireless, wired or hybrid network infrastructures. To realize the IEEE 802.11ad standard, we add various extra features to QualNet to support next generation Multi-Gbps WiFi. More specifically, the modules developed incorporate the following characteristics of the IEEE 802.11ad and hybrid DCN:

- The additional Modulation Coding Schemes (MCS) and their corresponding frame durations. In this context, as suggested in the standard and explained in section 2.4, the data frames are transmitted using MCS 24 while ACK frames use MCS 0.

- The IEEE 802.11ad MAC frame structure for each class.

- The PBSS-based network topology in which the PCP ensures i) Beacons transmission over the wired infrastructure and ii) the static association of DMG-STAs.
- The 4 wireless antennas deployed for each ToR.
- Both IEEE 802.3 (i.e., wired) and IEEE 802.11ad (i.e., wireless) protocols cohabit to design the hybrid DCN architecture.
- Cisco MSDC architecture is implemented.
- Wireless/Wireless and Wireless/Wired handover mechanisms are implemented.

The IEEE 802.11ad propagation parameters are set as in [8]. We assume that all the antennas have the same gain (i.e., transmitting and receiving) and the same transmission power which are respectively fixed to 0 dBm and 40 dBm. The Friis propagation model's parameter α is set to 2. Rx_Thr and CP_Thr are respectively set to -47 dBm and 10. Furthermore, according to IEEE 802.11ad specification, 4 wireless channels are available, with a bandwidth of 2.16GHz. Their running frequencies range from 57 GHz to 66 GHz.

To the best of our knowledge, this is the first implementation of IEEE 802.11ad in QualNet simulator.

Secondly, we built a Cisco MSDC's data center architecture. The geographic dimensions of the data center are 60m×60m forming a grid based infrastructure encompassing 256 racks. Each rack is composed of 20 servers and the overall infrastructure includes more than 5000 servers. Servers of the same rack are interconnected through a leaf switch (i.e., ToR). Each leaf is connected to 4 spine switches. As in [22], ToRs (i.e., leafs) are connected to servers via 1 Gbps links. Moreover, spine and leaf switches communicate through 10 Gbps links. Similarly to [8], we assume that the propagation delay of wired links is set to 2 μs. The noise factor and implementation loss values are respectively set to 10, and 5, as it is given by IEEE 802.11ad specification [23].

Finally, we implemented i) our wireless resource allocation algorithm `GC-HDCN` based on C++ language and CPLEX[2] solver and ii) the related strategies.

4.4.1.2 Simulation setup

Regarding the simulations setup, the traffic follows is a Constant Bit Rate (CBR) model characterized by i) the inter-arrival packet time of 6 μ-seconds and ii) the CBR packet size of 6214 Bytes. Note that the latter value is calibrated with respect to alleviate the fragmentation during the encapsulation process. In fact, the maximum size of IEEE 802.11ad frame is 7995 Bytes [23]. The volume of data transmitted in each communication follows a discrete uniform distribution taking values in $[3, 4]$ Gbit. We make use of UDP transport protocol to transmit the inter-rack traffic. On

[2] http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer

the other hand, the communicating servers of each transmission are chosen as follows: First the source server is uniform randomly selected among the set of racks deployed in the DCN. Then, the destination server is uniform randomly selected among the racks in which their set of WTUs located within the transmission range T_R of the source server's WTUs. We run the simulation for 100 communications. It is worth pointing out that each performance value of the implemented strategies is equal to the average of 6 simulations. Furthermore, our simulation results are always presented with confidence intervals corresponding to a confidence level of 95%.

4.4.2 Performance metrics

In order to evaluate the performances of GC-HDCN compared with the related approaches, we consider the following metrics:

1) $\mathbb{R}_L$: is the **R**esidual wire**L**ess traffic. It corresponds to the remaining amount of traffic to be transmitted over the ongoing wireless communications. It is straightforward to see that $\mathbb{R}_L$ evaluates the capacity of a channel allocation algorithm to carry out its traffic over the wireless infrastructure. Consequently, the higher the value of $\mathbb{R}_L$, the more the use of wireless resources is efficient.

2) $\mathbb{R}_D$: similarly to $\mathbb{R}_L$, this metric represents the **R**esidual wire**D** traffic. It corresponds to the remaining traffic of the ongoing communications to be transmitted over the wired infrastructure.

3) $\mathbb{D}$: is the cumulative delay of the network. In other words, it defines the cumulative transmission delay of all finished communications in the network. Let $\mathcal{N}$ denote the number of finished communications in the network and d_i the delay spent by a communication c_i to be transmitted. $\mathbb{D}$ is formulated as follows:

$$\mathbb{D} = \sum_{i=1}^{\mathcal{N}} d_i$$

4) $\mathbb{D}_{\mathbb{A}}$: is the Average Delay in the network, which defines the average transmission delay per traffic request.

4) $\mathbb{T}$: is the total throughput of the network. It corresponds to the cumulative transmission throughput of the traffic carried through the hybrid DCN.
Let c_i be the i^{th} finished communications in the network at the departure time l_i. Let v_i be the volume of traffic transmitted by the communication c_i. t_0 is the arrival time of the first communication c_0 in the network. If $\mathcal{N}$ represents the number of finished communications, $\mathbb{T}$ can be calculated as:

$$\mathbb{T} = \frac{\sum_{i=1}^{\mathcal{N}} v_i}{(l_{\mathcal{N}} - t_0)}$$

Table 4.1: Omni-WTU scenario: Average network metrics

	$\mathbb{D}_a$	$\mathbb{T}_a$
`GC-HDCN`(beamforming)	$6.96 \pm 0.25\%$	$178.50 \pm 20.44\%$
`GC-HDCN`	$9.18 \pm 1.36\%$	$156.33 \pm 21.42\%$
`Genetic-HDCN`	$30.72 \pm 6.89\%$	$117.96 \pm 23.89\%$
`Hungarian-HDCN`	$10.22 \pm 2.03\%$	$168.45 \pm 22.14\%$
`Wired-ECMP-HDCN`	$332.46 \pm 3.15\%$	$8.37 \pm 0.18\%$

4) $\mathbb{T}_{\mathbb{A}}$: is the Average Throughput in the network, which defines the average transmission throughput obtained per traffic request.

5) $\mathbb{S}_i$: denotes the Spatial Spectrum Reuse of channel i. $\mathbb{S}_i$ corresponds to the number of wireless communications which are simultaneously using the channel i. We recall that $i \in [1, 4]$ since the number of channels is equal to 4 for IEEE 802.11ad based networks.

6) $\mathbb{S}_{ia}$: is the average Spatial Spectrum Reuse of the i^{th} channel, $i \in \{1, .., 4\}$.

9) $\mathcal{T}_c$: represents the computation time of the decision algorithm.

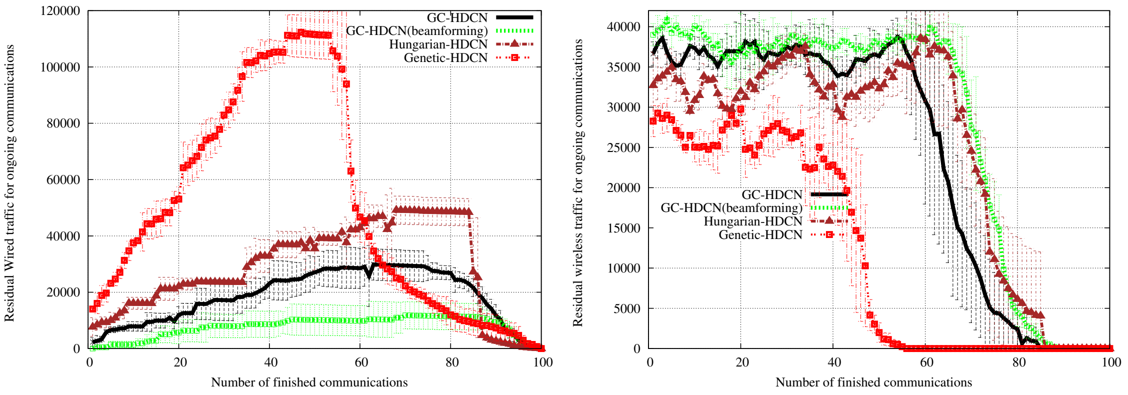
4.4.3 Simulation Results

To assess the efficiency of our proposal, we consider three main scenarios. First, Omni-Beam scenario, we compare the HDCN performance for both cases: i) omni-directional antennas and ii) beamforming technique. Secondly, Uniform-Load scenario, the communicating WTUs are equipped with directional switched-beam antennas, and traffic distribution follows a Poisson process. In third scenario, Real-Load scenario, we consider real workload traces of Facebook's DC.

4.4.3.1 Omni-Beam scenario

In the this scenario, similarly to [41], the transmission demands arrival follows a Poisson process with λ_A set to 4 communications per second. First, we evaluate our proposal by considering one-hop inter-rack communications where ToRs are equipped with omni-directional antennas, radiating signals in a uniform way, as in the related approaches, `Genetic-HDCN` [8] and `Hungarian-HDCN` [10]. Next, we resort to deploying switched-beam directional antennas on each ToR of the HDCN and we study the impact of the beamforming technique on the efficiency of our approach `GC-HDCN`. The objective is to prove the utility of beamforming mechanism to enhance the wireless resources usage and improve the network performance.

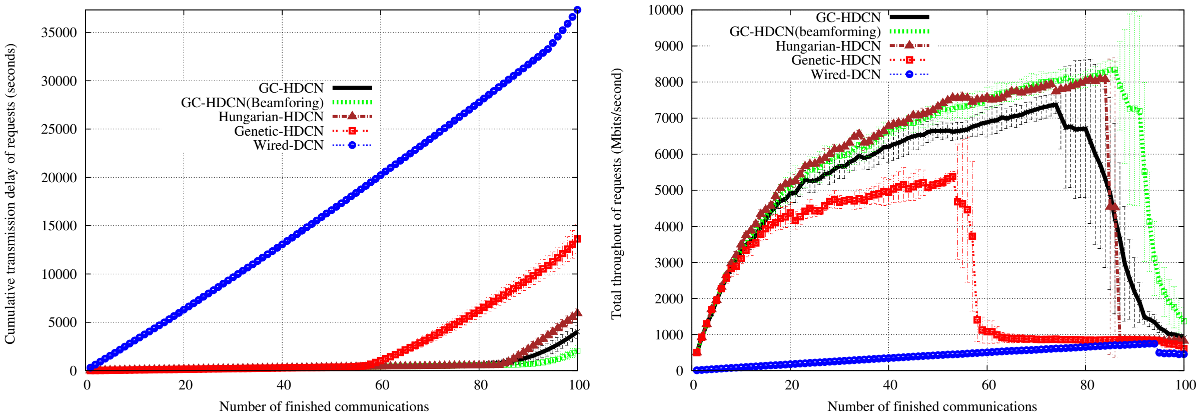
For both of the aforementioned deployment cases, we calculate, first, at each communication departure the amount of residual traffic (i.e., $\mathbb{R}_D$, $\mathbb{R}_L$) circulating in the network. In doing so,



(a) $\mathbb{R}_D$

(b) $\mathbb{R}_L$

Figure 4.2: Omni-Beam scenario: Wired & wireless residual traffic



(a) $\mathbb{D}$

(b) $\mathbb{T}$

Figure 4.3: Omni-Beam scenario: Network delay and throughput

we evaluate the ability of the resource allocation strategies to efficiently hand out ongoing communications, It is clear to see through Figure 4.2(a) and Figure 4.2(b) that `GC-HDCN` promotes wireless infrastructure. Note that for a given number of finished communications, a higher amount of residual wireless traffic with a lower proportion of wired traffic indicates that the use of wireless channels is enhanced. That is the case of our proposal which outperforms the related approaches. It is worth noting that such a strategy will guarantee a lower network delay and a higher throughput since hot wireless/wired links can be greatly alleviated.

To investigate the impact of the allocation strategies on the cumulative network performances, we evaluate the cumulative delay of the network, $\mathbb{D}$. The results are illustrated in Figure 4.3(a). It is straightforward to see that `GC-HDCN` ensures the lowest cumulative delay. Indeed, by the end

Table 4.2: Omni-Beam scenario: Average Spectrum Spatial Reuse

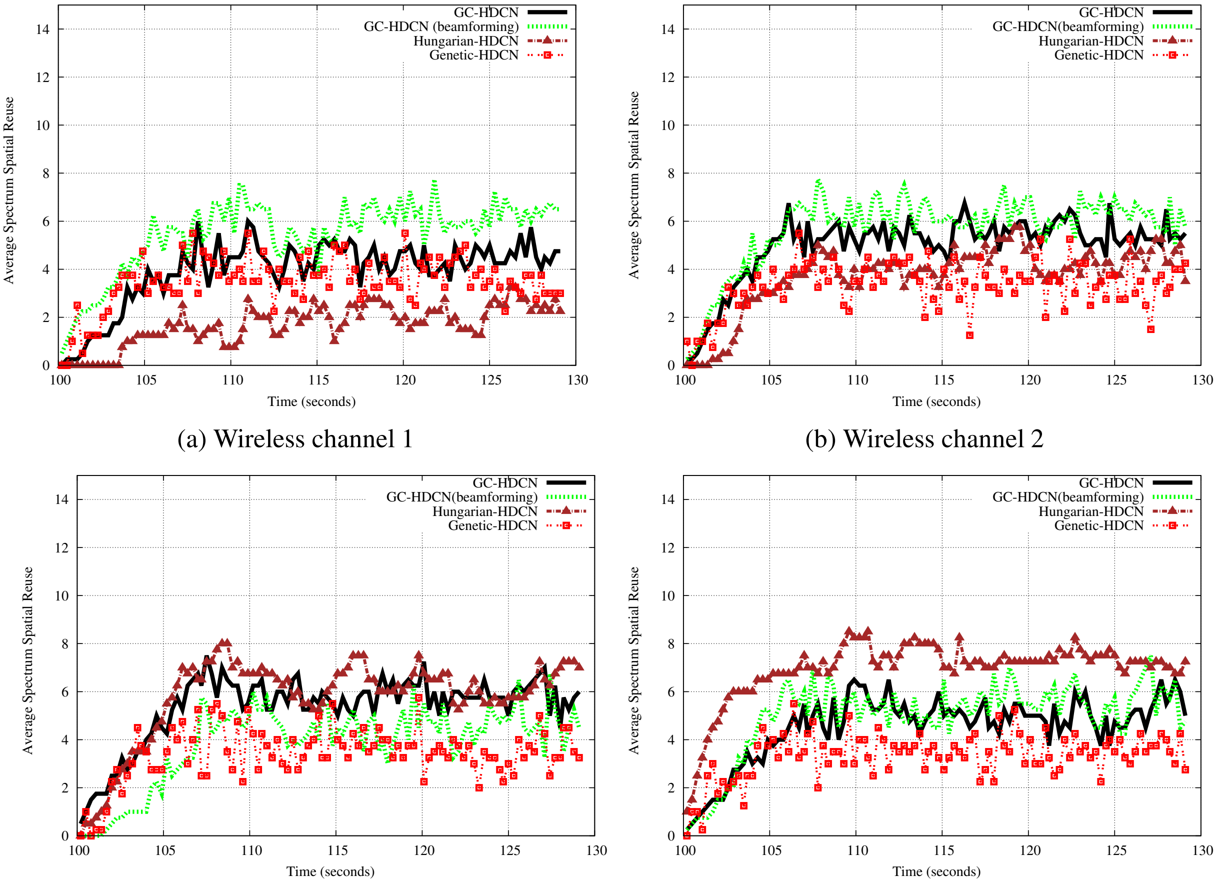
	GC-HDCN($beamforming$)	GC-HDCN	Genetic-HDCN	Hungarian-HDCN
$\mathbb{S}_{1a}$	$6.11 \pm 0.29\%$	$1.89 \pm 0.18\%$	2.75 ± 0.6	0.98 ± 0.10
$\mathbb{S}_{2a}$	$4.54 \pm 0.31\%$	$4.22 \pm 0.29\%$	2.89 ± 0.14	2.81 ± 0.22
$\mathbb{S}_{3a}$	$4.54 \pm 0.31\%$	$5.24 \pm 0.23\%$	2.80 ± 0.17	4.52 ± 0.30
$\mathbb{S}_{4a}$	$3.27 \pm 0.29\%$	$3.13 \pm 0.25\%$	2.97 ± 0.17	5.54 ± 0.27

of communications, our proposal reduces by respectively 26.21%, 67.77% and 88.59% the total network delay compared with Hungarian-HDCN, Genetic-HDCN and Wired-DCN. On the other hand, we notice that the use of beamforming technique enables our approach to further alleviate $\mathcal{D}$ by 57.99%. TABLE 4.1 illustrates the average transmission delay of the 100 communication demands. We remark that our approach improves $\mathbb{D}_{\mathbb{A}}$ by 70.11%, 10.17% and 94% compared respectively to Hungarian-HDCN, Genetic-HDCN and Wired-DCN. In addition, the use of beamforming mechanism further alleviates the average delay by 24.18%.

The obtained results corroborate those depicted in Figure 4.3(b) and confirm that our proposal maximizes the total network throughput. In fact, Figure 4.3(b) depicts the total network throughput, $\mathbb{T}$, according to the number of finished requests. It is worth pointing out that GC-HDCN achieves a higher total throughput which is improved respectively by 11.74%, 31.46% and 51.34% compared with Hungarian-HDCN, Genetic-HDCN and Wired-DCN related strategies. Besides, the throughput evolution is more noticeable when switched-beam antennas are deployed, in which case $\mathbb{T}$ is further enhanced by approximately 38.55%. Note that the total throughput decreases by the end of the simulation. This can by explained by the fact that wired communications leave lastly the network, which results in a high delay and consequently reduces the final throughput. Moreover, we notice that Hungarian-HDCN ensures by the beginning of simulations a higher throughput compared to GC-HDCN, because it basically allocates long alive requests on wired infrastructure. The latter take more time to leave the network, contrarily to our approach which minimizes the total traffic on wired network.

These results confirm those of the average network throughput presented through Table 4.1. It is clear to see that this metric is also enhanced as our strategy GC-HDCN improves $\mathbb{T}_a$ compared to almost the three related approaches. Moreover, thanks to beamforming technique, the average throughput is further enhanced by 12.42%.

In order to gauge the efficiency of the wireless resource use, we evaluate the Spatial Spectrum Reuse $\mathbb{S}_i$ for each channel w^i. We evaluate, in Figure 4.4, the Spatial Spectrum Reuse $\mathbb{S}_i$ for each channel w^i. We notice that our proposal makes use of all the wireless channels with the very close frequency values. Approximately, $\mathbb{S}_i$ of each channel is equal to 4. However, Hungarian-HDCN does not ensure the equilibrium of $\mathbb{S}_i$ among the channels. For instance, $\mathbb{S}_1$ is approximately equal



(a) Wireless channel 1

(b) Wireless channel 2

(c) Wireless channel 3

(d) Wireless channel 4

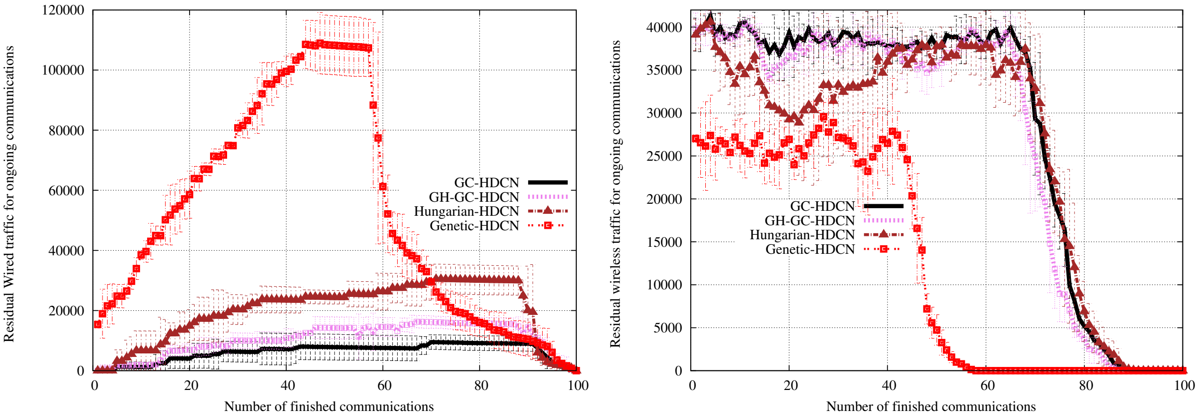
Figure 4.4: Omni-Beam scenario: Spatial Spectrum Reuse without beamforming

to 1 while $\mathbb{S}_4$ is approximately equal to 5. This imbalance on $\mathbb{S}_i$ impacts strongly the performance of the communications as illustrated in the above figures. Finally, we observe that the Spatial Spectrum Reuse of `Genetic-HDCN` is the worst one which consolidates the already presented results.

Table 4.2 shows that `GC-HDCN` ensures a high $\mathbb{S}_{ia}$ value varying between 3 and 6 for the four wireless channels, while it is equal to almost 2 for the `Genetic-HDCN` strategy. This weak channel re-utilization strongly impacts the performance of the communications as illustrated in the above results. Moreover, the high re-use of the spectrum by `GC-HDCN` is enhanced thanks to the beamforming technique.

Table 4.3: Average computation time $\mathcal{T}_c$

	GC-HDCN	GH-GC-HDCN
$\mathcal{T}_c$ (sec)	$169.9 \pm 36.17\%$	63.32 ± 5.63



(a) $\mathbb{R}_D$

(b) $\mathbb{R}_L$

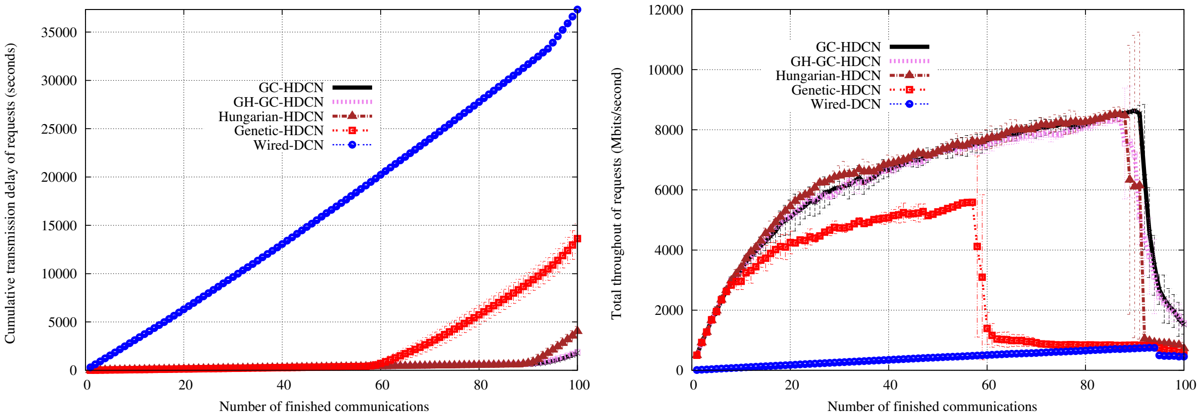
Figure 4.5: Uniform-Load: Wired & wireless residual traffic

Table 4.4: Uniform-Load: Average network metrics

	$\mathbb{D}_a$	$\mathbb{T}_a$
GC-HDCN	$6.96 \pm 0.25\%$	$178.50 \pm 20.44\%$
GH-GC-HDCN	$7.32 \pm 0.40\%$	$171.91 \pm 20.76\%$
Genetic-HDCN	$30.63 \pm 7.41\%$	$119.96 \pm 23.87\%$
Hungarian-HDCN	$8.98 \pm 1.47\%$	$173.37 \pm 21.88\%$
Wired-ECMP-HDCN	$332.46 \pm 3.15\%$	$8.37 \pm 0.18\%$

4.4.3.2 Uniform-Load scenario

Based on the results of the first scenario, it is straightforward to see that the beamforming technique improves the HDCN performance in terms of delay, throughput and spectrum reuse. Therefore, we deploy, in this scenario, only switched-beam antennas on ToRs. Moreover, we consider a uniform load pattern generated based on the Poisson process, similarly to [41], with λ_A set to 4 communications per second. We proceed as follows. First, we run experiments in order to gauge the efficiency of both our heuristic-based solution, GH-GC-HDCN, and exact solution, GC-HDCN, while evaluating the computation time. Second, we compare our both approaches to the related strategies Hungarian-HDCN, Genetic-HDCN and Wired-DCN.



(a) $\mathbb{D}$

(b) $\mathbb{T}$

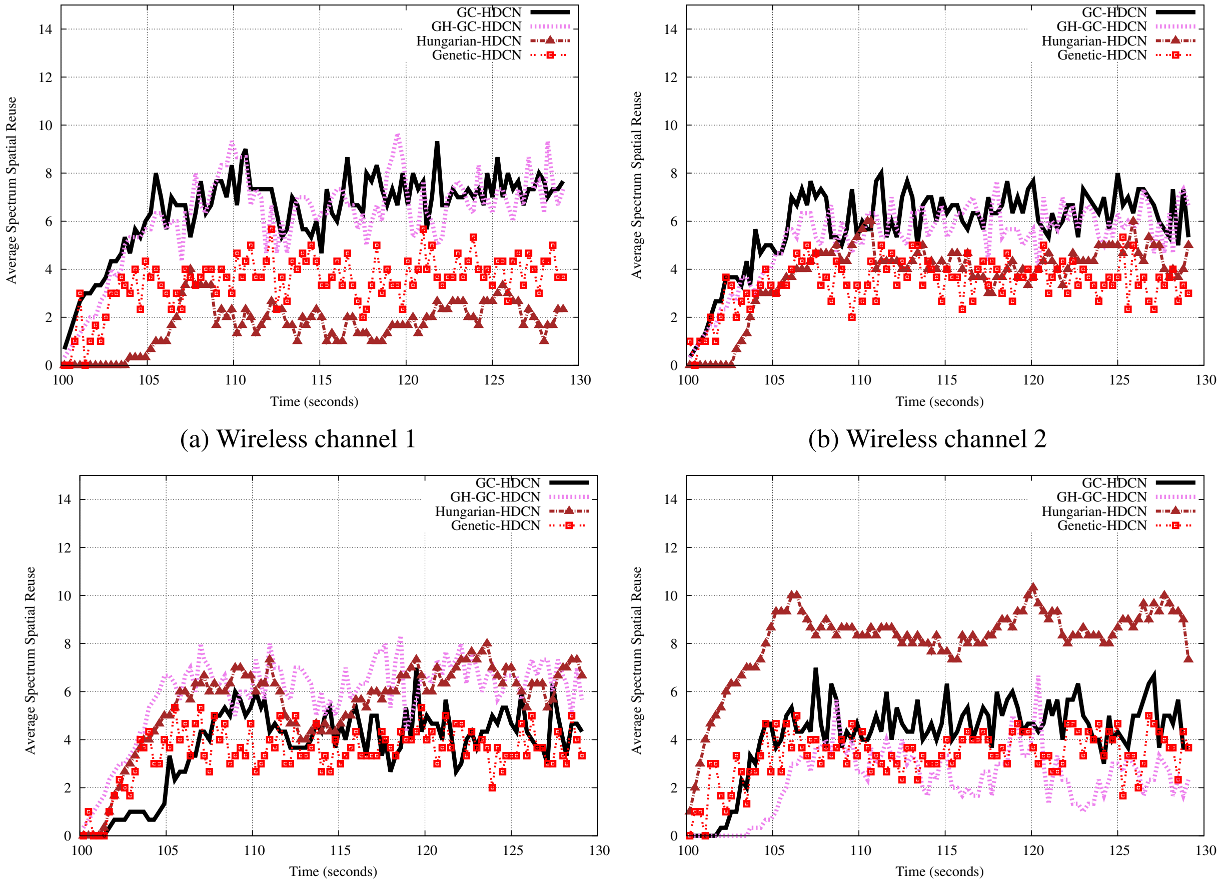
Figure 4.6: Uniform-Load: Network delay and throughput

Table 4.5: Uniform-Load: Average Spectrum Spatial Reuse

	`GC-HDCN`	`GH-GC-HDCN`	`Genetic-HDCN`	`Hungarian-HDCN`
$\mathbb{S}_{1a}$	$6.11 \pm 0.29\%$	$4.64 \pm 0.37\%$	$2.60 \pm 0.15\%$	1.0 ± 0.11
$\mathbb{S}_{2a}$	$4.54 \pm 0.31\%$	$6.92 \pm 0.30\%$	3.09 ± 0.15	2.63 ± 0.24
$\mathbb{S}_{3a}$	$4.54 \pm 0.31\%$	$2.64 \pm 0.26\%$	2.69 ± 0.17	4.27 ± 0.29
$\mathbb{S}_{4a}$	$3.27 \pm 0.29\%$	$0.91 \pm 0.11\%$	2.87 ± 0.15	6.60 ± 0.32

GH-GC-HDCN and GC-HDCN evaluation The computation time $\mathcal{T}_c$ of the column generation process is a key parameter of GC-HDCN since it simultaneously impacts: i) the solution quality, and ii) the complexity of the algorithm. Therefore, it is very crucial to evaluate the fastness level of GC-HDCN while guaranteeing a close-to optimal solution. In this stage, we run experiments for 100 inter-rack communication requests, and evaluate the average computation time of GC-HDCN, for both cases: i) GH-GC-HDCN, for which the pricing problem is generated based on the greedy GH+ heuristic, and ii) exact resolution of the pricing problem. The results of the average computation time, $\mathcal{T}_c$, are illustrated in Table 4.3. Deep experimental analysis show that when the size of the graph is greater to 10, the computation time of of the pricing problem using B&C algorithm explodes. Therefore, we set the threshold Thr_D to the value 10. It is straightforward to see that the use of the heuristic solution to generate new columns alleviates the time complexity.

Hereafter, we will compare the network performance of the above approaches to the related strategies.



(a) Wireless channel 1

(b) Wireless channel 2

(c) Wireless channel 3

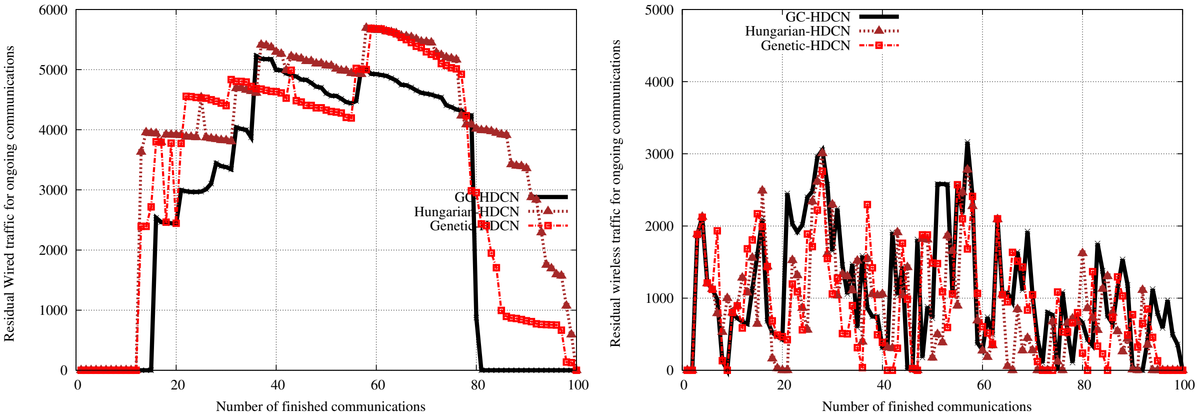
(d) Wireless channel 4

Figure 4.7: Uniform-Load: Spatial Spectrum Reuse

Comparison with related approaches Similarly to the above scenario antennas, we evaluate herein the residual resources as well as the cumulative throughput when beamforming mechanism is deployed in the HDCN. It is clear to see through Figure 4.5(a) and Figure 4.5(b) that both GC−HDCN and GH−GC−HDCN enhance the use of wireless infrastructure while reducing the traffic allocated through wired links.

Consequently, we notice that, as shown in Figure 4.6(a), GC−HDCN ensures the lowest cumulative delay ensured compared to the other strategies. Moreover, it is worth noting that our heuristic-based solution keeps a lower network delay compared to the other related strategies. These results corroborate with those of the average delay, illustrated in Table 4.4. In fact, we remark that both GC−HDCN and GH−GC−HDCN ensures the lowest value of $\mathbb{D}_a$.

Similarly, GC−HDCN further enhances the total throughput compared to the related approaches, thanks to the use of the switched-beam antennas. In fact, Figure 4.6(b) depicts the total network throughput, $\mathbb{T}$, according to the number of finished requests. It is obvious to see that our proposal



(a) $\mathbb{R}_D$

(b) $\mathbb{R}_L$

Figure 4.8: Real-Load: Wired & wireless residual traffic

Table 4.6: Real-Load: Average network metrics

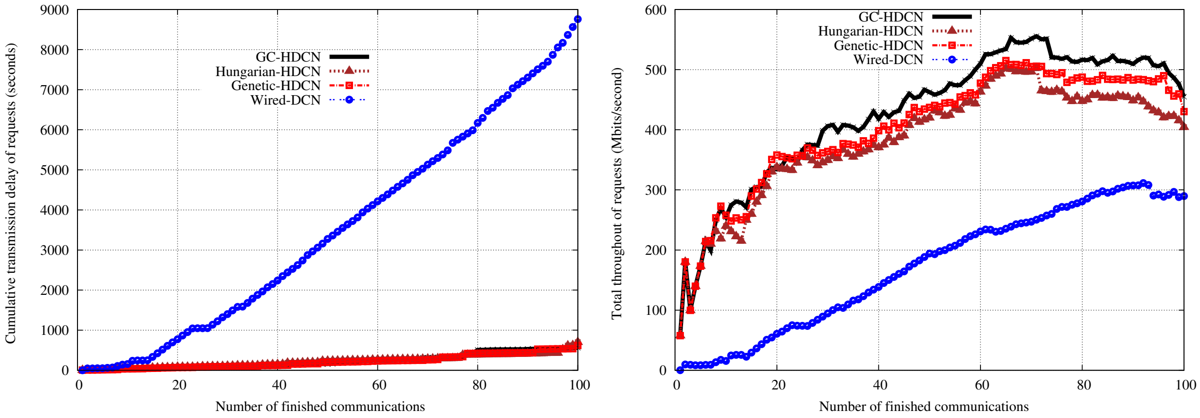
	$\mathbb{D}_a$	$\mathbb{T}_a$
GC-HDCN	$3.90 \pm 0.24\%$	$13.11 \pm 2.28\%$
Genetic-HDCN	$3.84 \pm 7.41\%$	$12.69 \pm 2.30\%$
Hungarian-HDCN	$4.18 \pm 0.14\%$	$12.12 \pm 2.28\%$
Wired-ECMP-HDCN	$57.45 \pm 4.46\%$	$3.19 \pm 0.13\%$

GH-GC-HDCN improves the throughput respectively by 53.33%, 67.29% and 70.83% compared with Hungarian-HDCN, Genetic-HDCN and Wired-DCN related strategies.

In order to further study the impact of our methods on resource usage, we evaluate the spectrum re-use per channel. Figure 4.7 shows that our approaches enhance in general the spectrum use for most of the channels. Table 4.2 shows that GC-HDCN ensures an average spectrum reuse $\mathbb{S}_{ia}$ varying between 6 and 3 for the four wireless channels, while it varies between 6 and 1 for our heuristic-based approach GH-GC-HDCN. Although the latter doesn't guarantee the same usage rate of different channels, it succeeds to enhance $\mathbb{S}_{ia}$ compared to Genetic-HDCN approach. Note that this efficient channel re-utilization strongly impacts the performance of the communications as illustrated in the above figures. Moreover, the strong re-utilization of the spectrum by our approach is enhanced thanks to the beamforming technique.

4.4.3.3 Real-Load scenario

In this scenario, we consider a real load traffic, dealing with the recent workload traces of **Facebook**'s DC [81]. In fact, Facebook monitoring system, fbflow, has collected, in 2015 for a period



(a) $\mathbb{D}$ (b) $\mathbb{T}$

Figure 4.9: Real-Load: Network delay and throughput

Table 4.7: Real-Load: Average Spectrum Spatial Reuse

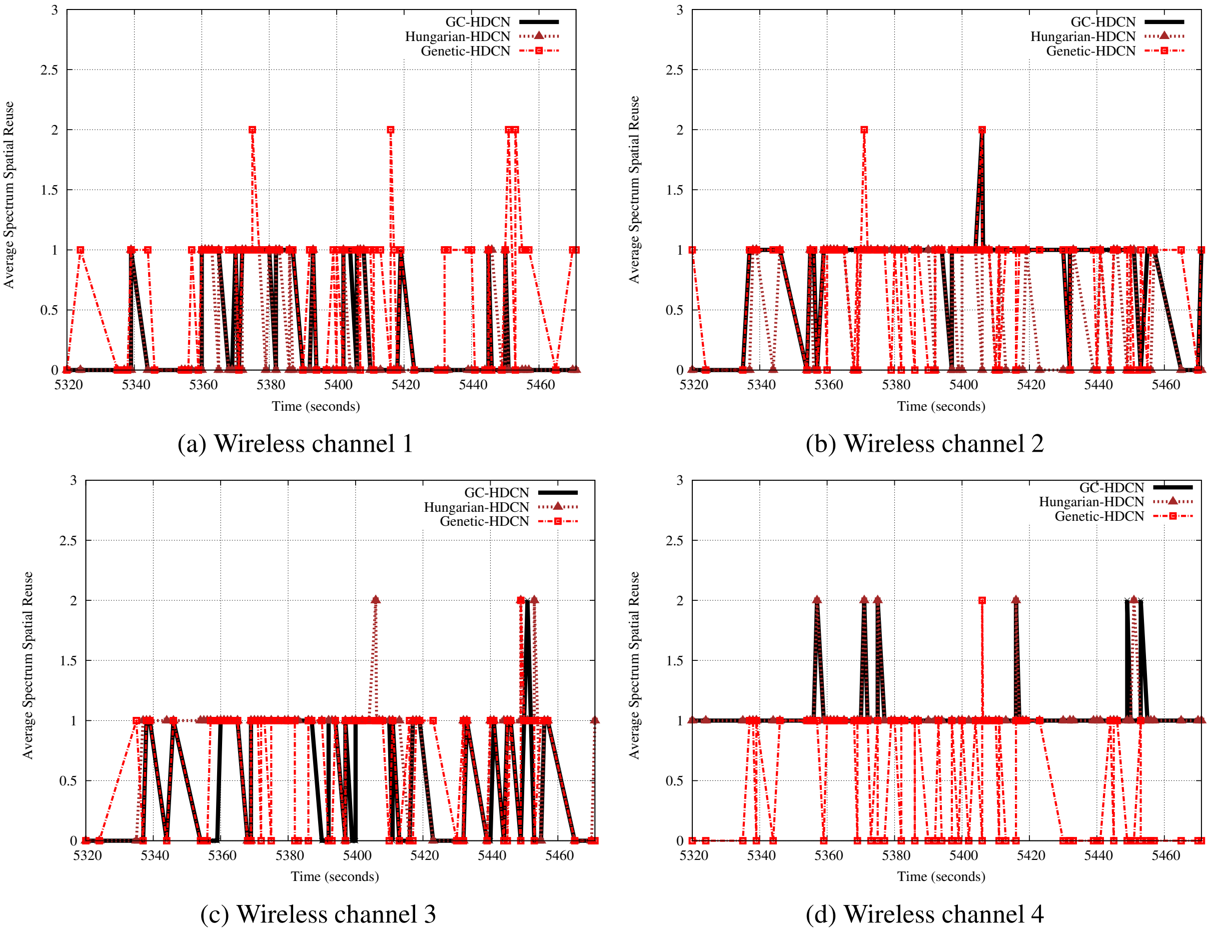
	`GC-HDCN`	`Genetic-HDCN`	`Hungarian-HDCN`
$\mathbb{S}_{1a}$	$0.39 \pm 0.03\%$	$0.65 \pm 0.03\%$	0.31 ± 0.02
$\mathbb{S}_{2a}$	$0.82 \pm 0.03\%$	0.78 ± 0.01	0.61 ± 0.03
$\mathbb{S}_{3a}$	$0.59 \pm 0.03\%$	0.64 ± 0.025	0.82 ± 0.03
$\mathbb{S}_{4a}$	$1.04 \pm 1.004\%$	0.63 ± 0.02	1.04 ± 0.004

of 24-hours, samples of traffic patterns inside the production clusters. Facebook has made accessible flow workload of some applications, namely: Hadoop, Web-servers, and Database. In our simulations, we consider the inter-rack traffic generated by Hadoop, since it is considered to be the heaviest [81].

Similarly, we proceed as follows. We have evaluated first the residual wireless and wired traffic transiting in the HDCN. We notice through Figure 4.8(a) and Figure 4.8(b) that our exact and heuristic based solutions, `GC-HDCN` and `GH-GC-HDCN`, promote the use of wireless infrastructure and further reduce the residual traffic on wired links, which alleviates bottlenecks in the HDCN.

In this regard, the total network delay of the Hadoop flows is reduced by our approach compared to the related strategies. Typically, Figure 4.9(a) shows that `GC-HDCN` impressively alleviates $\mathbb{D}$ by 19.8%, 8.9%, 93% compared respectively to `Hungarian-HDCN`, `Genetic-HDCN` and `Wired-DCN`. Consequently, the cumulative network throughput $\mathbb{T}$ is enhanced by our proposal with a rate of 11.3%, 5.4% and 36.31% compared to the same aforementioned methods.

Intuitively, the above results affirm those of the instantaneous spatial spectrum reuse. Actually, as depicted in Figure 4.10 and in Table 4.7, our approach ensures a higher spectrum reuse compared



(a) Wireless channel 1

(b) Wireless channel 2

(c) Wireless channel 3

(d) Wireless channel 4

Figure 4.10: Real-Load: Spatial Spectrum Reuse

to `Genetic-HDCN`. Whereas, `Hungarian-HDCN` shows comparable $\mathbb{S}_{ia}$ values to our method.

4.5 Conclusion

In this chapter, we tackled the problem of traffic congestion in data center networks. To do so, we augmented the CISCO MSDC wired data center with wireless infrastructure based on IEEE 802.11ad in order to minimize the congestion and enhance network performances. Additionally, we have deployed the 2D beamforming technique in order to alleviate interference effects and leverage wireless infrastructure use. Besides, we proposed a new wireless channel allocation mechanism, named `GC-HDCN`, in a Hybrid data center network. We formulated our NP-hard problem as a Graph Coloring and we made use of Column Generation and Branch-and-Price algorithms to resolve it. Accordingly, `GC-HDCN` has two variants: i) an exact variant making use of the exact resolution of the pricing problem, and ii) a heuristic variant, `GH-GC-HDCN`, based on a Greedy heuristic to find new potential columns, while alleviating computation time. Our objective is to minimize traffic congestion by maximizing the use of wireless channels. Extensive simulations with QualNet simulator, for both uniform and real Facebook's workload traces, show that our proposal enhances data center performances and outperforms the most prominent related strategies in terms of: i) total network delay, ii) total network throughput, and iii) spectrum spatial reuse.

The obtained `GC-HDCN` results are however restricted to the case of single-hop communications, where racks have to be placed in the same coverage area. Actually, in a real DC, distant servers can transmit traffic flows, and, thus, multi-hop communications are required in HDCN. To deal with this limitation, we will address, in the next chapter, the problem of joint routing and channel assignment for multi-hop inter-rack communications in HDCN. Specifically, we will propose an online novel approach that sequentially computes for each communication request the hybrid (wireless/wired) routing path while assigning channels.

Chapter 5

Joint online routing and channel allocation in HDCN

Contents

5.1 Introduction

In the previous chapter, we have proposed a novel wireless channel allocation in HDCN to carry one-hop communications while enhancing network performance. Unfortunately, in spite of the impressive results of our proposal compared to the related strategies, it is restricted to the case where the communicating racks are in the same transmission range. Therefore, `GC-HDCN` can not deal with multi-hop communications. Moreover, our literature review presented in Section 3.3 and

Section 3.4 shows that while few researches have dealt with channel allocation problem in single hop, rare are those which addressed the issue of jointly routing and allocating wireless channel for multi-hop communications in HDCN.

In this chapter, we will tackle the problem of online joint routing and channel allocation in HDCN. The main focus is to harness jointly wireless and wired interfaces to enhance the data center network capabilities in term of bandwidth. In doing so, the end-to-end delay and the congestion of wired infrastructure are minimized. To achieve our goal, we put forward a Centralized Controller (CC) scheduler that monitors the traffic and jointly computes the flow routes and channel assignment. Indeed, we propose an advanced Joint Routing and Channel Assignment algorithm for HDCN (JRCA-HDCN), which harvests both wired and wireless infrastructures. The key idea behind JRCA-HDCN is to take into consideration both the i) length of IP queues (waiting delay) in each relay node and ii) level of wireless interferences (retransmission delay) among intra-flow (successive wireless links) and inter-flows. Assuming a data flow from source $\mathcal{S}$ to destination $\mathcal{D}$, JRCA-HDCN computes the optimal hybrid path that reduces the end-to-end delay. Note that JRCA-HDCN is an **online** approach that processes sequentially each incoming communication requests as it arrives. Our problem is formulated as a Minimum Weight Perfect Matching (MWPM). We perform extensive network simulations in QualNet simulator while considering the full protocol stack (from application to physical layers), to gauge the performance of JRCA-HDCN algorithm. The obtained results are compared to those of the related strategies, and to our previous proposal GC-HDCN dealing one-hop communications.

The remainder of the chapter is organized as follows. In Section 5.2, we present our HDCN model and formulate the joint routing and channel allocation problem within HDCN. Afterwards, Section 5.3 will describe the details of our proposal JRCA-HDCN. Simulation environment and performance evaluation will be presented in Section 5.4. Finally, Section 5.5 will conclude the chapter.

5.2 Problem Formulation

In this section, we will, first, define the model of inter-rack wireless/wired network. Then, we will formulate the joint routing and channel assignment problem in HDCN based on Minimum Weight Perfect Matching (MWPM) model.

5.2.1 Hybrid Data Center Network Model

We define a Wireless/Wired Transmission Unit (WTU), denoted by W_i, as a group of servers in a rack sharing a set of wireless beamforming antennas and a gigabit wired switch. Each W_i is equipped with 4 IEEE 802.11ad transceivers/antennas (i.e., orthogonal channels) denoted by $\{w_i^1, w_i^2, w_i^3, w_i^4\}$ and one Top of Rack switch (ToR) based on IEEE 802.3 denoted by w_i^5. Note that the communications between $\{W_i\}$ (i.e., inter-rack) are ensured by both: i) a wireless infrastructure (through $\{w_i^1, w_i^2, w_i^3, w_i^4\}$) and/or ii) a wired infrastructure (through w_i^5).

We model the HDCN as an undirected graph $\mathcal{G} = (V(\mathcal{G}), E(\mathcal{G}))$. Each node $v_i \in V(\mathcal{G})$ corresponds to one WTU W_i. An edge $e \in E(\mathcal{G})$ between two nodes v_i and v_j exists if and only if they can communicate in full-duplex among all the wireless channels of IEEE 802.11ad while assuming the absence of interferences. We make use of the Friis signal transmission model. This is motivated by the fact that obstacles are non existent in the data center environment and radio antennas are deployed on the top of racks. The receiving signal power sent by w_i^k to w_j^k is equal to :

$$P_r(i,j,k) = P_t + G(\theta(i,j,k)) + 20\log_{10}\left(\frac{\eta}{4\pi d}\right)^{\alpha} - \tau - \psi \tag{5.2.1}$$

where i) P_t is transmitting signal power, ii) $G(\theta(i,j,k))$ is the gain of transmitting and receiving antennas and $\theta(i,j,k)$ refers to the azimuth angle between antennas, iii) η (meter) is the wavelength, iv) d (meter) is separating distance between w_i^k and w_j^k, v) α represents the path loss effects, and vi) τ and ψ are respectively the noise factor and the implementation loss fixed in IEEE 802.11ad standard [23]. Note that a signal transmitted on channel k from w_i^k is successfully received at w_j^k if i) $P_r(i,j,k) \geq Rx\_Thr$ where Rx_Thr is a predefined threshold representing the receiver hardware sensitivity.

The Signal to Interference Noise Ratio between transmitter w_i^k and destination w_j^k on the channel k is equal to:

$$SINR(i,j,k) = \frac{P_r(i,j,k)}{\sum_{m \neq i} P_i(m,j,k)} \tag{5.2.2}$$

$P_i(m,j,k)$ is the interference power received at antenna w_j^k and caused by w_m^k on the beam used in the communication initiated by w_i^k. It is worth noting that w_i^k succeeds to communicate with w_j^k (i.e., without interference) if and only if $SINR(i,j,k)$ and $SINR(j,i,k)$ (i.e., ACK reception) are at least equal to CP_Thr. The latter is a hardware constant of the transceiver.

Formally, we model the interference between two communication links $e = (w_i^k, w_j^k)$ and $e' = (w_m^k, w_n^k)$ as follows : i) transmitter, w_i^k of e interferes with receiver w_n^k of e' or ii) transmitter w_m^k of e' interferes with receiver w_j^k of e.

We distinguish two kinds of interferences in HDCN: i) intra-flow and ii) inter-flow interferences. Intra-flow interferences are caused by two successive links belonging to the same path and simultaneously using an identical wireless channel. Thanks to the beamforming technique, the non-successive links are not interfering. Inter-flow interferences are caused by active links belonging to different paths and transmitting over the same wireless channel. In order to avoid the intra-flow interferences, WTU cannot receive and transmit simultaneously on the same channel. On the other hand, inter-flows interferences are minimized by selecting wireless links with minimal cost in term of retransmission delay.

Given the static topology of racks in the HDCN, we initialize the $SINR$ table with the signal-to-noise ratio values between all the racks for different antennas orientations (i.e., beams). Then, entries in this table are opportunistically refreshed, during the ongoing wireless traffic transmissions. In fact, measurements of $SINR$ of active racks at different antenna orientations can be retrieved by the CC, using the wired infrastructure.

5.2.2 Joint Routing and Channel Assignment problem

The joint routing and channel assignment problem in HDCN consists in computing, for a given communication from wireless/wired transmission unit W_s to W_d, the optimal **hybrid** path satisfying i) elimination of intra-flow interferences by considering wireless channel allocation of all successive hops, ii) minimization of inter-flows interferences by considering the retransmission cost, iii) minimization of waiting delay by considering the length of IP queues in the path (wired and/or wireless).

Note that the above optimal path is hybrid (wireless and/or wired). Unfortunately, the undirected weighted graph $\mathcal{G} = (V(\mathcal{G}), E(\mathcal{G}))$ does not include i) wired links and ii) wireless channel links. In fact, an edge in $\mathcal{G}$ between W_i and W_j is the fusion of the four wireless channel links. For this reason, we propose to extend $\mathcal{G}$ to include the missing links. To do that, we extend the Edmonds-Szeider ($\mathcal{ES}$) [82] node expansion technique to generate a new graph denoted by $\hat{\mathcal{G}} = \left(\hat{V}\left(\hat{\mathcal{G}}\right), \hat{E}\left(\hat{\mathcal{G}}\right)\right)$. $\hat{\mathcal{G}}$ supports simultaneously **wired** and multi-channel **wireless** links, as detailed in sub-section 5.2.2.1. The problem of optimal hybrid path from W_s to W_d is formulated as Minimum Weight Perfect Matching (MWPM) problem in $\hat{\mathcal{G}}$ as detailed in sub-section 5.2.2.2. In fact, the path is built by the concatenation of matching links in $\hat{\mathcal{G}}$. Note that each edge in $\hat{\mathcal{G}}$ is associated to exactly one interface (wireless channel or wired).

5.2.2.1 Edmonds-Szeider Expansion

We recall that each wireless/wired transmission unit W_i is equipped with 4 wireless interfaces denoted by $\{w_i^1, w_i^2, w_i^3, w_i^4\}$ and the wired ToR switch interface denoted by w_i^5. Using Edmonds-Szeider expansion ($\mathcal{ES}$) [82], G is transformed to the new expanded graph $\hat{\mathcal{G}}$. The latter is generated using the following operations:

1. Each node $v_i \in V(\mathcal{G}) \backslash \{W_s, W_d\}$ is expanded into into 12 sub-nodes as follows:
   - $8 = 2\times4$ wireless sub-nodes referring to the wireless channels: $\{v_i^1, v_i^{1'}, v_i^2, v_i^{2'}, v_i^3, v_i^{3'}, v_i^4, v_i^{4'}\}$.
   - 2 wired sub-nodes $\{v_i^5, v_i^{5'}\}$.
   - 2 extra sub-nodes $\{v_i^g, v_i^{g'}\}$ which are used to connect all the above sub-nodes.
2. Each pair of sub-nodes $(v_i^k, v_i^{k'})$ is attached with zero-cost **internal** link as illustrated in Figure 5.1.

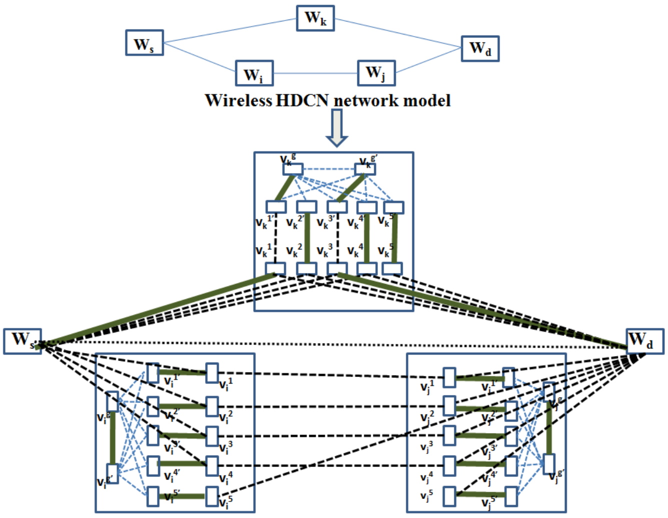


Figure 5.1: Illustration of Edmonds-Szeider expansion

3. Each edge $e_i \in E(\mathcal{G})$ is expanded into 4 (i.e., number of channels) **exterior** links denoted by $\{\hat{e}_i^1, \hat{e}_i^2, \hat{e}_i^3, \hat{e}_i^4\}$. If e_i is attaching v_m and v_n that implies each exterior link $\hat{e}_i^k$, $k \in \{1, .., 4\}$ will attach $\hat{v}_m^k$ and $\hat{v}_n^k$ (analogous sub-nodes in term of wireless channel) as shown in Figure 5.1.

4. Once $\mathcal{ES}$ expansion technique converges and hence $\hat{\mathcal{G}}$ is partially generated, the latter is augmented by connecting both W_s and W_d with all their 1-hop wireless neighbors sub-nodes.

5. Finally, each wired sub-node in $\hat{\mathcal{G}}$ is directly attached to the destination W_d through an exterior wired edge. Indeed, the latter represents the two-hop wired OSPF path in the MSDC architecture and our objective is to reach the final destination W_d. Therefore, it is straightforward to see that our optimal path cannot contain two consecutive wired links and it is not judicious to link the intermediate nodes in the path using wired interfaces.

Note that $\hat{\mathcal{G}}$ is **weighted** undirected graph where the cost of each **exterior** link $\hat{e} \in \hat{E_E}\left(\hat{\mathcal{G}}\right)$ is equal to:

$$\mathcal{C}(\hat{e}) = \frac{1}{\mathcal{D}(\hat{e})} \cdot [1 + \alpha \cdot \mathcal{F}(\hat{e}) + \frac{\varrho \sum_{\bar{e} \in \mathcal{I}(\hat{e})} \mathcal{R}(\bar{e})}{\alpha}] \tag{5.2.3}$$

where iii) $\mathcal{D}(\hat{e})$ is the data rate of the $\hat{e}$'s sending extremity, ii) $\mathcal{I}(\hat{e})$ is the set of all **active** wireless interfering links with $\hat{e}$, iii) $\alpha = max\{1, |\mathcal{I}(\hat{e})|\}$ is a coefficient reflecting the number of interfering links if they exist, iv) $\mathcal{F}(\hat{e})$ represents the sum of residual and requested traffic volumes (wired or wireless) in IP queue, v) $\mathcal{R}(\hat{e})$ denotes the residual traffic in the IP queue of an interfering link $\hat{e}$, and vi) ϱ is the maximum number of frame retransmissions and fixed by IEEE 802.11ad standard to 7. We assume that if $\hat{e}$ is wired interface then $|\mathcal{I}(\hat{e})| = 0$. It is worth pointing out that the cost of a link incarnates the transmission delay of its residual (wireless or wired) traffic and the resulting retransmission delays (wireless) caused by/on interfering links. Moreover, weights are dynamically computed as the $SINR$ is instantaneously refreshed, as explained above.

5.2.2.2 Minimum Weight Perfect Matching formulation

Now $\hat{\mathcal{G}} = \left(\hat{V}\left(\hat{\mathcal{G}}\right), \hat{E}\left(\hat{\mathcal{G}}\right)\right)$ is fully constructed (i.e., vertices, edges and cost) in which the optimal hybrid path between W_s to W_d will be searched. We formulate the joint routing and channel assignment in HDCN as a Minimum Weight Perfect Matching problem. In fact, computing the minimum cost alternating-hybrid path is equivalent to find the **minimum** weight **perfect** matching in the expanded graph $\hat{\mathcal{G}}$. A perfect matching in $\hat{\mathcal{G}}$ is defined as a subset of links $\tilde{E} \subseteq \hat{E}\left(\hat{\mathcal{G}}\right)$ such as each vertex $v \in \hat{V}\left(\hat{\mathcal{G}}\right)$ has **exactly one** incident link $\tilde{e} \in \tilde{E}$. In doing so, finding the perfect matching in $\hat{\mathcal{G}}$ guarantees that two successive links in the path cannot make use the same channel (i.e., alternation). Therefore, the obtained path is free of intra-flow interferences. Moreover, we seek for the path with the minimum total cost (see equation 5.2.3) in order to minimize both waiting delay (length of IP queues) and retransmissions delay (inter-flows interferences).

It is worth noting that computing the minimum weight alternating-hybrid path is equivalent to computing the minimum weight perfect matching in the expanded graph $\hat{\mathcal{G}}$. Indeed, by expanding the initial graph G to $\hat{\mathcal{G}}$, each node in G was exploded into **even number** of sub-nodes with zero-cost internal edges (see Figure 5.1). Consequently, it is straightforward to find a zero-cost matching within each expanded node by exclusively using internal links. Since the source W_s and destination W_d are not expanded in $\hat{\mathcal{G}}$, the perfect matching will inevitably have at least two external links: one **coming from** the source W_s and the second **going to** the destination W_d. Besides, each one of them has necessarily a sub-node (i.e., wireless or wired exploded node) extremity. Each expanded node in $\hat{\mathcal{G}}$ would have either two or none external links in the perfect matching. Therefore, the set of external links belonging to the perfect matching will necessary construct the path connecting the source to the destination with every relay node is visited exactly once (no loop). Consequently, the minimum weight perfect matching corresponds to the solution such as the cumulative cost of external links is minimal. In return, all the sub-nodes not belonging to the above optimal hybrid path are trivially matched through their zero-cost internal links. Finally, to obtain the final optimal path, each selected expanded node in $\hat{\mathcal{G}}$ is contracted to a single node (i.e., come-back) and the unmatched exterior links will be removed.

Formally, for each subset $\tilde{V} \subseteq \hat{V}\left(\hat{\mathcal{G}}\right)$, each link $e \in \hat{E}\left(\hat{\mathcal{G}}\right)$ from u to v satisfying both conditions i) $u \in \tilde{V}$ and ii) $v \in \hat{V}\left(\hat{\mathcal{G}}\right) \backslash \tilde{V}$, is in the set of boundary links of $\tilde{V}$ denoted by $\delta(\tilde{V})$. We denote by $\mathcal{B}$ the set of all subsets of $\tilde{V}$ of odd cardinality containing at least three nodes. We refer to these subsets by **blossoms**. It is worth pointing out that a blossom is recursively composed of **pseudo-nodes** which may be either nodes in $\tilde{V}$ or blossoms in $\mathcal{B}$.

The MWPM problem based on the Primal and Dual Edmond's linear programming statements are sequentially formulated hereafter:

$$\begin{array}{ll} & \texttt{Primal Problem} \\ \texttt{minimize} & \sum_{\hat{e}\in\hat{E}(\hat{\mathcal{G}})} \mathcal{C}(\hat{e}) \cdot x(\{\hat{e}\}) \\ \texttt{subject to:} & x(\delta(\{\hat{v}\})) = 1, \ \forall \hat{v} \in \hat{V}\left(\hat{\mathcal{G}}\right) \\ & x(\delta(\hat{B})) \geq 1, \ \forall \hat{B} \in \mathcal{B} \\ & x(\{\hat{e}\}) \geq 0, \ \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right) \end{array}$$

$$\begin{array}{ll} & \texttt{Dual Problem} \\ \texttt{maximize} & \sum_{\hat{v}\in\hat{V}(\hat{\mathcal{G}})} y_{\hat{v}} + \sum_{\hat{B}\in\mathcal{B}} y_{\hat{B}} \\ \texttt{subject to:} & slack(\hat{e}) \geq 0, \ \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right) \\ & y_{\hat{B}} \geq 0, \ \forall \hat{B} \in \mathcal{B} \end{array}$$

Note that $x(\{\hat{e}\})$ in the primal problem is binary ($\{0,1\}$) variable indicating whether the link $\hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right)$ is matched or not and $\mathcal{C}(\hat{e})$ is the cost value defined in equation 5.2.3. The first constraint in the primal problem ensures that each node in $\hat{V}\left(\hat{\mathcal{G}}\right)$ will be matched exactly once. However, it is not sufficient to claim that a perfect matching could be obtained. In fact, in each blossom $\hat{B} \in \mathcal{B}$ of odd cardinality n, there are at most $(n-1)$ pseudo-nodes that may be trivially matched using internal edges forming $\hat{B}$. Therefore, according to the second constraint, at least one pseudo-node in $\hat{B}$ should be obviously matched with a link $\hat{e} \in \delta(\hat{B})$.

In the dual problem, $slack(\hat{e})$, denoting the reduced cost of an edge $\hat{e} = (u,v) \in \hat{E}\left(\hat{\mathcal{G}}\right)$, is defined as follows:

$$slack(\hat{e}) = \mathcal{C}(\hat{e}) - y_u - y_v - \sum_{\hat{B}\in\mathcal{B}:\hat{e}\in\delta(\hat{B})} y_{\hat{B}} \tag{5.2.4}$$

According to the first constraint, slack values must be positive for all edges. The second constraint implies that blossoms should always keep positive dual values. Given a dual solution $\bar{Y}$, an edge is called **tight** if its slack is equal to zero. A blossom $\hat{B} \in \mathcal{B}$ is called **full**, if $x(\delta(\hat{B}))$ is equal to 1.

We define the complementary slackness conditions for the primal and dual problems as follows: i) for each edge $\hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right)$, if $x(\{\hat{e}\}) = 0$ then $\hat{e}$ is tight ($slack(\hat{e}) > 0 \Longrightarrow x(\{\hat{e}\}) = 0$) and ii) for each blossom $\hat{B} \in \mathcal{B}$, $y_{\hat{B}} > 0$ implies that $\hat{B}$ is full ($y_{\hat{B}} > 0 \Longrightarrow x(\delta(\hat{B})) = 1$).

Note that a given perfect matching is optimal (i.e., minimal) if its dual solution satisfies the aforementioned conditions. It is straightforward to see that all perfect matchings of $\hat{\mathcal{G}}$ correspond to feasible solutions of MWPM problem since the incidence vector of any perfect matching satisfies the linear system. To reach the optimal solution, the idea is to maintain a feasible dual vector and a integer-valued primal vector which corresponds to a matching. These vectors will be gradually updated until reaching optimal perfect matching.

Algorithm 11: `JRCA-HDCN` pseudo-algorithm

```
Inputs: $\hat{\mathcal{G}}$
Output: $\mathcal{M}_{opt}$
$\mathcal{M}_0$ ←Initial-Matching($\hat{\mathcal{G}}$)
$\bar{Y}_0$ ←Initial-Dual-Values($\hat{\mathcal{G}}$)
$Perfect \leftarrow false, i \leftarrow 1, \mathcal{M}_i \leftarrow \mathcal{M}_0, \hat{\mathcal{G}}_i, \leftarrow \hat{\mathcal{G}}, \bar{Y}_i \leftarrow \bar{Y}_0$
repeat
    ($\mathcal{M}_{tmp}$, $\hat{\mathcal{G}}_{tmp}$) ← Primal-operations-stage ($\hat{\mathcal{G}}_i$, $\mathcal{M}_i$)
    if $\mathcal{M}_{tmp}$ is perfect then
        $\mathcal{M}_{opt} \leftarrow \mathcal{M}_{tmp}$
        $Perfect \leftarrow true$
    else
        $tight \leftarrow false$
        $\bar{Y}_{tmp} \leftarrow \bar{Y}_i, i \leftarrow i + 1$
        $\mathcal{M}_i \leftarrow \mathcal{M}_{tmp}, \hat{\mathcal{G}}_i \leftarrow \hat{\mathcal{G}}_{tmp}$
        repeat
            $\bar{Y}_i$ ← Dual-updates-stage($\hat{\mathcal{G}}_i$)
            if $\bar{Y}_i = \bar{Y}_{tmp}$ then
                $tight \leftarrow true$
        until $tight = true$;
until $Perfect = true$;
```

5.3 Proposal: JRCA-HDCN

In this section, we will detail our proposal named **Joint Routing and Channel Allocation strategy in Hybrid Data Center Network** (`JRCA-HDCN`) to resolve the formulated problem in the previous section. Our proposal is based on the last variant of **Edmond's Blossom V** algorithm [83]. The main specificity of this version consists in combining the use of i) multiple-tree search approaches described in **Blossom IV** variant and ii) sophisticated data structures in order to reach a polynomial convergence time equal to $\mathcal{O}(|\hat{V}\left(\hat{\mathcal{G}}\right)| \times |\hat{E}\left(\hat{\mathcal{G}}\right)|^2)$ as proven in [83].

`JRCA-HDCN` proceeds as follows. First, ***i) Initialization stage*** generates the first matching $\mathcal{M}_0$ of $\hat{\mathcal{G}}$ and calculates the dual values vector $\bar{Y}_0$. Then, ***ii) Primal operations stage*** is performed by executing sequentially and repetitively augment, grow, shrink and expand operators in aim to augment the matching until the perfect matching (optimal solution) is reached or stability of matching. If stability, then `JRCA-HDCN` proceeds the ***iii) Dual updates stage*** until at least one tight edge appears. Next, our strategy comes back to the ***Primal operations stage***. `JRCA-HDCN` is summarized in the pseudo Algorithm 11. In the following, we will detail each stage.

5.3.1 Initialization stage

Initially, we consider an empty matching $\mathcal{M}_0$ for which $x(\{\hat{e}\}) = 0$ for each edge $\hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right)$. The dual value $y_{\hat{v}}$ for each node $\hat{v} \in \hat{V}\left(\hat{\mathcal{G}}\right)$ is set to $\frac{1}{2}\min_{\hat{e}\in\delta(\hat{v})}\{\mathcal{C}(\hat{e})\}$. By doing so, we ensure that $slack(\hat{e})$ cannot be negative.

5.3.2 Primal operations stage

In order to perform primal operations, our algorithm builds at each iteration an alternating tree rooted at an unmatched node. To this end, each node $\hat{v} \in \hat{V}\left(\hat{\mathcal{G}}\right)$ is assigned one label $\mathcal{L}(\hat{v}) \in \{+, -, \emptyset\}$. The label $+$ is, first, assigned to each unmatched node that will form the root of an alternating tree $\mathcal{T}$. Each $+$ labeled node is connected to $-$ labeled one using one **tight unmatched** edge. Note that node $-$ labeled $\hat{v}$ node is necessarily the parent of a $+$ labeled one using a **tight matched** edge. Finally, $\emptyset$ labeled nodes are called free and represent the matched nodes that do not belong to any alternating tree.

Complexity of Blossom algorithm strongly depends on the way that trees are explored during both primal and dual updates processes. Three main approaches for tree processing can be adopted: i) single tree, ii) multiple trees with fixed dual change and iii) multiple trees with variable dual change. We seek for the approach leading to a short augmenting path in fewer operations. It has been proven in [83] that the efficient approach consists in combining both single strategy and multiple strategy with fixed dual change. Indeed, based on some experiments, we came to realize that the matching of the last nodes requires usually the higher time. Therefore, we propose to match the first 90% of the nodes using the single approach and the remaining 10% with the multiple approach.

Primal updates are operations performed on the alternating trees using only tight edges, as illustrated in Figure 5.2. The aim behind this stage is to find an augmenting path and hence increase the matching cardinality. To do so, basically four primal operations are iteratively performed:

1. **`AUGMENT`**: This operator is performed when a tight edge connects two nodes both labeled with $+$ and belonging to different trees. Reversing the matching along the edges between the roots of the two trees produces an augmenting path. Note that `AUGMENT` is the key operation of primal updates since it increases by 1 the cardinality of the matching.

2. **`GROW`**: This operator grows tree $\mathcal{T}$ by adding two tight edges. It is performed on node $\hat{v}_1 \in \mathcal{T}$ if $\mathcal{L}(\hat{v}_1) = +$ and there exists a free node $\hat{v}_2$ (i.e., $\mathcal{L}(\hat{v}_2) = \emptyset$), matched to another free node $\hat{v}_3$, such that the link between $\hat{v}_1$ and $\hat{v}_2$) is tight. In such case, $\mathcal{T}$ is grown by the link between $\hat{v}_1$ and $\hat{v}_2$ and link between $\hat{v}_2$ and $\hat{v}_3$. The labels of $\hat{v}_2$ and $\hat{v}_3$ are respectively set to $-$ and $+$.

3. **`SHRINK`**: This operator checks whether a cycle of an odd number of nodes and tight edges

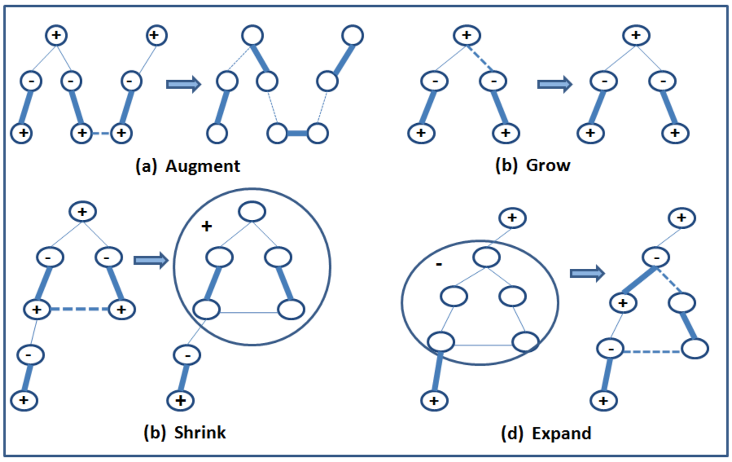


Figure 5.2: Primal updates

exists in a tree $\mathcal{T}$. The blossom exists if and only if two $+$ labeled nodes $\hat{v}_1$ and $\hat{v}_2$ are connected with a tight edge. SHRINK operator substitutes the blossom by a single node.

4. **EXPAND**: This operator expands each shrunk blossom $\hat{B}$ node with labeled $-$ if its dual value is equal to 0. In doing so, the dual value cannot be negative and hence ensure the duality constraint in the dual problem formulation.

As in Blossom V implementation, we grow trees in depth-first search way in order to reduce computation time. We put forward, also, a specific order giving priority to AUGMENT then GROW operators. Both SHRINK and EXPAND are executed only when AUGMENT and GROW fail. Indeed, AUGMENT is the unique operation that increases the current matching.

5.3.3 Dual updates stage

The main objective of this stage is to generate new tight edges so that new primal operations can be performed again on trees. To do so, some specific updates are applied to the dual vector $\bar{Y}$ in such way that the objective function of the dual problem increases while satisfying the duality constraints. The idea is to update the dual value $y_{\hat{v}}$ of each non free node $\hat{v} \in \mathcal{T}$ by an amount $\epsilon_{\mathcal{T}} \geq 0$ as following: $y_{\hat{v}} = y_{\hat{v}} + \epsilon_{\mathcal{T}}$ if $\mathcal{L}(\hat{v}) = +$ and $y_{\hat{v}} = y_{\hat{v}} - \epsilon_{\mathcal{T}}$ if $\mathcal{L}(\hat{v}) = -$.
$\epsilon_{\mathcal{T}}$ is defined in such way that dual vector $\bar{Y}$ should remain feasible during each dual adjustment stage. Typically, $\epsilon_{\mathcal{T}}$ is set to the maximum value simultanously satisfying the following constraints:

$$\begin{cases} \epsilon_{\mathcal{T}} \leq slack(\hat{v}_1, \hat{v}_2) & \text{If } (\mathcal{L}(\hat{v}_1), \mathcal{L}(\hat{v}_2)) = (+, \emptyset) \\ & \text{and } \hat{v}_1 \in \mathcal{T} \\ \epsilon_{\mathcal{T}} + \epsilon_{\mathcal{T}'} \leq slack(\hat{v}_1, \hat{v}_2) & \text{If } (\mathcal{L}(\hat{v}_1), \mathcal{L}(\hat{v}_2)) = (+, +) \\ & \text{and } \hat{v}_1 \in \mathcal{T}, \hat{v}_2 \in \mathcal{T}', \mathcal{T} \neq \mathcal{T}' \\ \epsilon_{\mathcal{T}} \leq slack(\hat{v}_1, \hat{v}_2)/2, & \text{If } (\mathcal{L}(\hat{v}_1), \mathcal{L}(\hat{v}_2)) = (+, +) \\ & \text{and } \hat{v}_1, \hat{v}_2 \in \mathcal{T} \\ \epsilon_{\mathcal{T}} \leq y_{\hat{v}} & \hat{v}_1 \text{ is blossom and } \mathcal{L}(\hat{v}) = - \\ & \text{and } \hat{v} \in \mathcal{T} \\ \epsilon_{\mathcal{T}} - \epsilon_{\mathcal{T}'} \leq slack(\hat{v}_1, \hat{v}_2) & \text{If } (\mathcal{L}(\hat{v}_1), \mathcal{L}(\hat{v}_2)) = (+, -) \\ & \text{and } \hat{v}_1 \in \mathcal{T}, \hat{v}_2 \in \mathcal{T}', \mathcal{T} \neq \mathcal{T}' \end{cases} \tag{5.3.5}$$

Accordingly, for each tree T_i, ϵ_{T_i} is set to min{$\epsilon_{i,1}$, $\epsilon_{i,2}$, $\epsilon_{i,3}$, $\epsilon_{i,4}$, $\epsilon_{i,5}$} with:

$$\begin{aligned} \epsilon_{i,1} &= min\{slack(u, v) : (u, v) = (+, \emptyset) \in E',\ u \in T_i\} \\ \epsilon_{i,2} &= min\{slack(u, v)/2 : (u, v) = (+, +) \in E',\ u \in T_i,\ v \in T_j\} \\ \epsilon_{i,3} &= min\{slack(u, v)/2 : (u, v) = (+, +) \in E',\ u \in T_i\} \\ \epsilon_{i,4} &= min\{y_u : u \in B,\ l(u) = -,\ u \in T_i\} \\ \epsilon_{i,5} &= min\{slack(u, v)/2 : (u, v) \in E',\ u \in T_i,\ v \in T_j\} \end{aligned}$$

where T_i and T_j denote two alternating trees.

It is straightforward to see that after each dual update at least one primal operation will be performed on the tree. Indeed, updating dual values by an amount of $\epsilon_{i,1}$ leads to a GROW operation in T_i, while an adjustment with $\epsilon_{i,2}$ results in at least one augmenting path between T_i and T_j. Similarly, if ϵ_T=$\epsilon_{i,3}$, then there is at least one odd cycle that will be shrinked, while it will be expanded if ϵ_T=$\epsilon_{i,3}$. Note that the goal of the latter expansion is to keep feasible the second constraint of MWPM problem. Finally, an update with $\epsilon_{i,5}$ may not necessarily result in a primal operation.

5.4 Performance Evaluation

In this section, we will report the performance of our JRCA-HDCN algorithm by performing a series of detailed simulations. We start with describing the stages of our implementation and environment set up. Afterwards, we define the performance metrics we consider to evaluate our strategy. Finally, we analyze the results and discuss the effectiveness of our proposal compared to the most relevant related strategies found in literature.

5.4.1 Simulation Environment and Methodologies

5.4.1.1 Experiment Design

We make use of QualNet[1], an event driven network simulation platform based on C++ language, and widely used by the network research community. To realize IEEE 802.11ad standard, we

[1] http://www.scalablenetworkors.com/products/Qualnet/

integrate new features to QualNet to support next generation Multi-Gbps WiFi.

We set the propagation parameters and rate table based on the IEEE 802.11ad. We assume that all the antennas have the same transmission power which is fixed to 10 dBm. We configure the QualNet physical layer with the free-space propagation model, by setting the Friis parameter α to 2. Rx_Thr and CP_Thr values are respectively set to -78 dBm and 10. Furthermore, 4 wireless channels are available according to IEEE 802.11ad specification, with a bandwidth of 2.16 GHz and running frequencies ranging from 57 GHz to 66 GHz.

To deploy beamforming technique, we associate 4 switched-beam antennas, composed of 8 beams, to each ToR. Besides, we build a Cisco MSDC's data center, containing 256 racks [22], in which we: i) use OSPF protocol for traffic routing and ii) implement ECMP protocol in order to balance the load over the wired network. Each rack contains 20 servers and the overall infrastructure includes more than 5000 servers. The geographic dimensions are 60m×60m. Servers of the same rack are interconnected through a leaf switch (i.e., ToR). Each leaf is connected to 4 spine switches. As in [22], ToRs (i.e., leafs) are connected to servers via 1 Gbps links. Moreover, spine and leaf switches communicate through 10 Gbps links. Similarly to [8], we assume that the propagation delay of wired links is set to 2 μs. The noise factor and implementation loss values are respectively set to 10, and 5, as it is given by IEEE 802.11ad specification [23].
Finally, we implemented i) our joint routing and channel allocation algorithm `JRCA-HDCN` based on C++ language and Boost[2] library and ii) the related strategies.

5.4.1.2 Simulation setup

Regarding the simulations setup, the traffic follows a Constant Bit Rate (CBR) model for which we set the inter-arrival packet time to 6 μ-seconds and the CBR packet size to 6214 Bytes. Note that the latter value is calibrated in a way that no fragmentation occurs during the encapsulation process. In fact, the maximum size of IEEE 802.11ad frame is 7995 Bytes [23]. We make use of UDP transport protocol to transmit the inter-rack traffic. The volume of data to transmit for each communication follows a random uniform distribution between 3 and 4 Gbytes. We run the simulation for 100 transmission demands. The confidence interval is fixed to 95%.

5.4.2 Performance metrics

We consider several metrics to evaluate our purposes:

1. $\mathbb{D}$: is the cumulative delay of the network. It defines the cumulative transmission delay of all the finished communications in the network. Let $\mathcal{F}$ denote the number of finished communications in the network and d_i the delay spent by a communication c_i to be transmitted. $\mathbb{D}$ is formulated as follows: $\mathbb{D} = \sum_{i=1}^{\mathcal{F}} d_i$

[2] http://www.boost.org/

2. $\mathbb{D}_{\mathbb{A}}$: is the average delay of the network. It defines the average transmission delay of all the finished communications in the network.

3. $\mathbb{T}$: is the total throughput of the network. Let c_i be the i^{th} finished communication in the network at the departure time l_i, v_i the volume of traffic transmitted by c_i, t_0 the arrival time of the first flow. For $\mathcal{N}$ finished communications, $\mathbb{T}$ is given by: $\mathbb{T} = \frac{\sum_{i=1}^{\mathcal{N}} v_i}{(l_{\mathcal{N}} - t_0)}$

4. $\mathbb{T}_{\mathbb{A}}$: is the average throughput of the network. It corresponds to the average transmission throughput per request of the traffic carried through the hybrid DCN.

5. $\mathbb{S}_{ia}$: is the average Spatial Spectrum Reuse of the i^{th} channel, $i \in \{1, .., 4\}$.

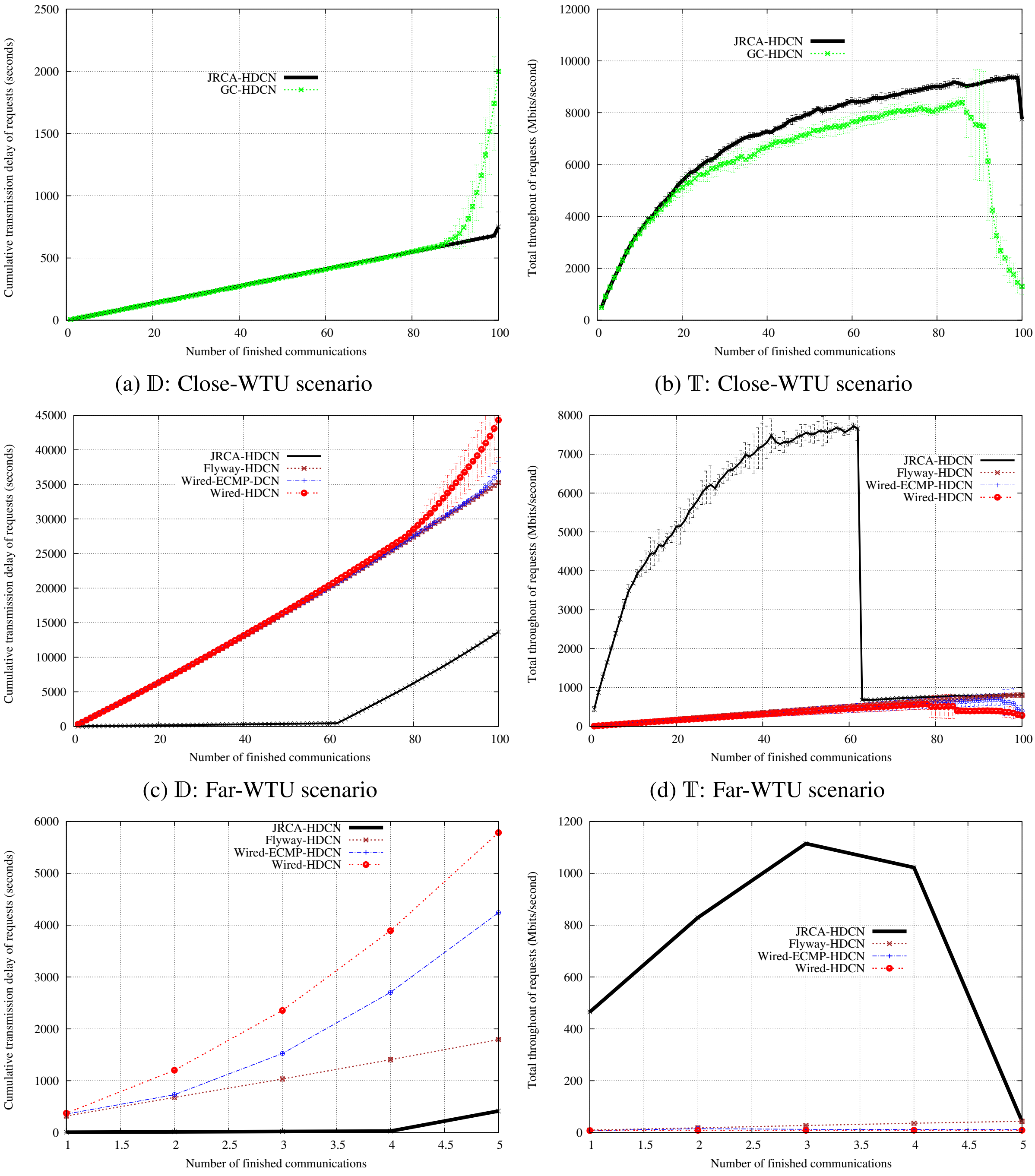
5.4.3 Simulation Results

To assess the efficiency of our proposal, we consider four main scenarios. In the first scenario, Close-WTU scenario, the communicating WTUs are close to each other, while in the second, Far-WTU scenario, the communicating WTUs are not placed in the same transmission range. In the third, Hotspot scenario, we deal with the specific configuration of `Flyway-HDCN` where many hotspot links are generated. In the above three scenarios, we generate transmission demands by following a Poisson process, similarly to [41], with an arrival rate λ_A equal to 4 communications per second. The fourth scenario, Real-Load scenario, we consider the recent real workload of **Facebook**'s DC [81].

5.4.3.1 Close-WTU and Far-WTU scenarios

In the Close-WTU scenario, we evaluate our proposal by considering the same scenario as our prior one-hop communication approach `GC-HDCN` [24], where the source and destination WTUs are in the same transmission range. The objective is to prove the necessity of multi-hop communications in the case of wireless resources shortage and the resort to the wired network which offers lower bandwidth. In the Far-WTU scenario, we deal with the far communicating racks which are not placed within the same transmission range and consequently flows need to be carried by multi-hop paths. We randomly choose the destination server based on a uniform distribution among the racks in which the WTUs can not communicate in one-hop with the sending server. We compare the efficiency of our strategy with the related methods: i) `Flyway-HDCN`, ii) `Wired-ECMP-HDCN` and iii) `Wired-HDCN` (i.e., without ECMP).

For the both aforementioned scenarios, we first evaluate the cumulative delay of the network, $\mathbb{D}$. The results are illustrated in Figure 5.3(a) for the Close-WTU scenario and in Figure 5.3(c) for the Far-WTU scenario. It is straightforward to see that `JRCA-HDCN` ensures the lowest cumulative delay. Indeed, by the end of communications, our proposal reduces $\mathbb{D}$ by 62.51% compared to `GC-HDCN`, which proves that multi-hop transmissions enhance the HDCN performance



(a) $\mathbb{D}$: Close-WTU scenario

(b) $\mathbb{T}$: Close-WTU scenario

(c) $\mathbb{D}$: Far-WTU scenario

(d) $\mathbb{T}$: Far-WTU scenario

(e) $\mathbb{D}$: Hotspot scenario

(f) $\mathbb{T}$: Hotspot scenario

Figure 5.3: Total network Delay and Throughput

for close WTUs. Moreover, our approach reduces $\mathbb{D}$ by 61.21%, 61.93% and 66.94% compared to `Flyway-HDCN`, `Wired-ECMP-HDCN` and `Wired-HDCN`. Table 5.1 illustrates the average trans-

Table 5.1: Average network metrics

	$\mathbb{D}_a$	$\mathbb{T}_a$
JRCA-HDCN	$35.09 \pm 8.25\%$	$151.18 \pm 26.33\%$
Flyway-HDCN	$330.79 \pm 2.59\%$	$8.70 \pm 0.063\%$
Wired-ECMP-HDCN	$331.39 \pm 2.69\%$	$8.62 \pm 0.12\%$
Wired-HDCN	$339.93 \pm 4.73\%$	$8.056 \pm 0.30\%$

Table 5.2: Average Spectrum Spatial Reuse

	JRCA-HDCN	Flyway-HDCN
channel 1	$16.93 \pm 1.08\%$	$1.08 \pm 0.12\%$
channel 2	$16.48 \pm 1.07\%$	$1.022 \pm 0.14\%$
channel 3	$15.47 \pm 1.14\%$	$0.67 \pm 0.14\%$
channel 4	$15.87 \pm 1.20\%$	$0.55 \pm 0.11\%$

mission delay of the 100 communication demands. We remark that our approach improves $\mathbb{D}_{\mathbb{A}}$ by 89.39%, 89.41% and 89.67% compared respectively to Flyway-HDCN, Wired-ECMP-HDCN and Wired-HDCN.

Figure 5.3(b) and Figure 5.3(d) depict the total network throughput, $\mathbb{T}$, according to the number of finished requests, for respectively Close-WTU and Far-WTU scenarios. It is worth pointing out that JRCA-HDCN achieves the highest total throughput than the related approaches. In fact, by the end of transmissions, our proposal improves the throughput respectively by 83.20%, 2.35%, 52.12% and 65.81% compared to GC-HDCN, Flyway-HDCN, Wired-ECMP-HDCN and Wired-HDCN strategies. Note that the total throughput decreases by the end of the simulation. This is because wired communications leave lastly the network, which results in high delay and consequently reduces the final throughput.

The obtained results corroborate the previous ones depicted in Figure 5.3(a) and Figure 5.3(c) and confirm that our proposal alleviates network delay, and hence enhances network performance. Additionally, this confirms the results of the average network throughput presented through Table 5.1. It is clear to see that the latter is also enhanced as our strategy JRCA-HDCN improves the average throughput by approximately 94% compared the three routing approaches. In fact, our approach carries flows on both wireless and wired infrastructure while taking into account the link capacity and the waiting delays.

In order to gauge the efficiency of the wireless resource use, we evaluate the average Spatial Spectrum Reuse $\mathbb{S}_{ia}$ for each channel w^i. Table 5.2 shows that JRCA-HDCN ensures a high $\mathbb{S}_{ia}$ value varying between 15 and 16 for the four wireless channels, while it is equal to almost 1 for the

Flyway-HDCN strategy. This weak channel re-utilization strongly impacts the performance of the communications as illustrated in the above results. Moreover, the high re-use of the spectrum by JRCA-HDCN is enhanced thanks to the beamforming antenna.

5.4.3.2 Hotspot scenario

In the Far-WTUs scenario, we noticed that Flyway-HDCN strategy does not achieve good performance since it is conceived to deal with HDCN with many hotspots. Therefore, we study in this scenario a highly congested HDCN. To this end, we simultaneously carry 5 (i.e., number of interfaces per rack) traffic demands incoming from the same source WTU, denoted WTU_i, to a uniform randomly chosen destination. In doing so, the ToR of WTU_i becomes oversubscribed, and hence potential hotspots appear.

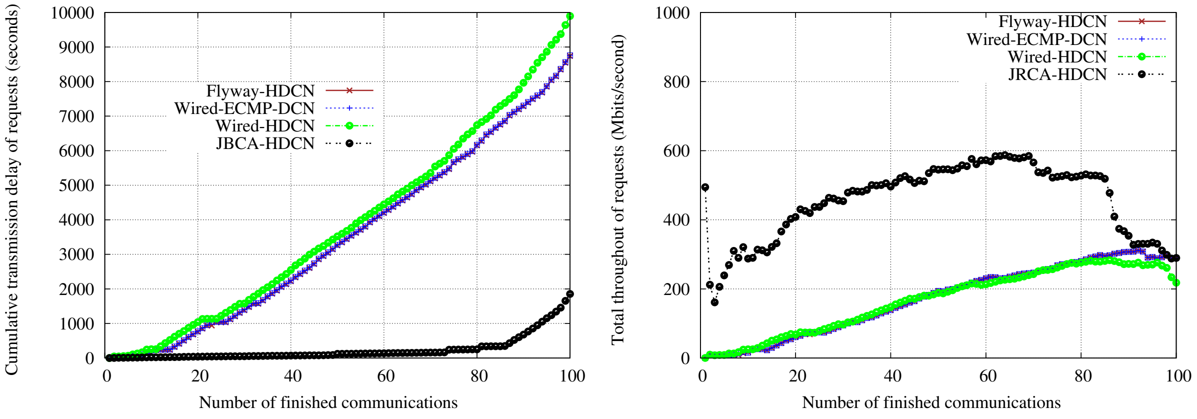
We study the behavior of JRCA-HDCN and the strategies: i) Flyway-HDCN, ii) Wired-HDCN and iii) Wired-ECMP-HDCN, towards the oversubscribed links. Figure 5.3(e) illustrates the cumulative delay in the DCN by the end of each communication. We notice that Wired-ECMP-HDCN and Wired-HDCN dramatically increase the network delay. Flyway-HDCN relieves hotspot effects and decreases the delay compared to the the classical wired strategies. Our approach JRCA-HDCN alleviates the network delay by 76.84% compared to Flyway-HDCN thanks to the 4 available wireless interfaces. Similarly, JRCA-HDCN clearly enhances the total network throughput compared to Flyway-HDCN as shown through Figure 5.3(f).

5.4.3.3 Real-Load

In this scenario, we consider the flow traces recently generated by **Altoona Facebook**'s data center [81]. In fact, Facebook monitoring system, fbflow, has collected, in 2015 for a period of 24-hours, samples of traffic patterns inside the production clusters. Facebook has made accessible flow workload of some applications, namely: Hadoop, Web-servers, and Database. In our simulations, we consider of the inter-rack traffic generated by Hadoop, since it is considered to be the heaviest [81].

We consider our online approach JRCA-HDCN, where each single Hadoop flow is routed as it arrives. We compare the performance of JRCA-HDCN to the related online approaches i) Flyway-HDCN, ii) Wired-ECMP-HDCN and iii) Wired-HDCN. We first evaluate the cumulative delay of the network, $\mathbb{D}$. The results are illustrated in Figure 5.4(a). It is straightforward to see that JRCA-HDCN importantly reduces the delay compared to the related online strategies. Indeed, by the end of communications, our proposal drastically alleviates the total network delay by 78%, 77% and 81.12% compared to respectively Flyway-HDCN, Wired-ECMP-HDCN and Wired-HDCN.

These results corroborate those of the average transmission delay, illustrated in Table 5.3. We remark that our online method JRCA-HDCN ensures the lowest average delay compared to the related



(a) $\mathbb{D}$

(b) $\mathbb{T}$

Figure 5.4: Real-Load scenario: Hadoop cluster in Facebook

Table 5.3: Average network metrics: Real-Load

	$\mathbb{D}_a$	$\mathbb{T}_a$
`JRCA-HDCN`	$3.45 \pm 11.84\%$	$19.45 \pm 26.33\%$
`Flyway-HDCN`	$56.91 \pm 35.17\%$	$3.19 \pm 0.52\%$
`Wired-ECMP-HDCN`	$57.45 \pm 19.15\%$	$3.18 \pm 0.49\%$
`Wired-HDCN`	$64.52 \pm 4.38\%$	$3.17 \pm 0.11\%$

Table 5.4: Average Spectrum Spatial Reuse: Real-Load

	`JRCA-HDCN`	`Flyway-HDCN`
$\mathbb{S}_{1a}$	$4.01 \pm 0.38\%$	$0.27 \pm 0.06\%$
$\mathbb{S}_{2a}$	$3.34 \pm 0.52\%$	$0.105 \pm 0.01\%$
$\mathbb{S}_{3a}$	$2.08 \pm 0.52\%$	$0.35 \pm 0.08\%$
$\mathbb{S}_{4a}$	$2.29 \pm 0.39\%$	$0.75 \pm 0.23\%$

methods. This decrease in the network delay comes with the benefits of enhancing the throughput. In fact, as shown in Figure 5.4(b), for our online method `JRCA-HDCN`, $\mathbb{T}$ is roughly two to three times higher than that of the related online strategies.

Furthermore, we evaluate, through Table 5.4 the average Spatial Spectrum Reuse $\mathbb{S}_{ia}$ for each channel w^i. We notice that while `JRCA-HDCN` makes use of all the wireless channels with the same frequency, our proposals in general enhance $\mathbb{S}_{ia}$.

5.5 Conclusion

In this paper, we addressed the problem of multi-hop communications and wireless channel assignment in hybrid data center networks. To alleviate congestion effects, we proposed to augment the conventional wired DCNs by wireless infrastructure (IEEE 802.11ad standard) while minimizing interferences by deploying 60 GHz 2D beamforming antennas. Moreover, we evaluated the efficiency of our proposal in a large-scale data center architecture based on the CISCO's Massively Data Center model. We formulated our problem as a Minimum Weight perfect Matching and we made use of the recent variant of Edmond's Blossom algorithm to obtain the optimal solution. Extensive simulations conducted within QualNet simulator show that our approach outperforms the most related strategies in terms of end-to-end delay, throughput and spectrum spatial reuse.

In the next chapter, we will deal with the batch joint routing and channel allocation problem in order to handle the batched arrivals of communications to the HDCN. Indeed, flow demands in real DCs such as Facebook and Google are almost arriving in batch. Therefore, sequentially processing communications in an online way does not ensure an efficient resource assignment. To this end, we will propose a novel joint batch-routing and channel allocation approach, so that we further optimize the wireless resource usage and enhance HDCN performance.

Chapter 6

Joint batch routing and channel allocation in HDCN

Contents

6.1 Introduction

In this chapter, we address the joint routing and channel allocation issue for batched flow requests within HDCN. Our main concern is to harness both the wireless and wired interfaces to carry a set of inter-rack communications, so that to enhance the DCN performance. To do so, the routing and wireless channels allocation are optimized. The issue of jointly routing and allocating wireless channels for multi-hop communications in HDCN, while considering **hybrid** paths, has been rarely addressed. Although this issue has been heavily studied in the literature in the context of Mesh networks, the related approaches ensure only fully wireless paths which is unfortunately not applicable to HDCN. While we process, in our previous chapter, each single communication flow in an online way, we focus, in this contribution, on carrying the flows in a **batch** mode for a better use of HDCN resources. In fact, the arrival order closely impacts the HDCN performance. Therefore, we deal with the **J**oint **B**atch **R**outing and **C**hannel **A**ssignment problem (`JBRC`) in HDCN, to handle the batched arrivals of communication flows. In doing so, the communications, arriving during a specific time window, are queued together and their processing is delayed to the following time window. Specifically, we put forward a Centralized Controller (CC) that monitors the traffic and jointly computes the flow routes and channel assignment.

We formulate `JBRC` using an advanced Multi-Commodity Flow (MCF) model, where each commodity corresponds to a communication demand. The objective of `JBRC` is to find for each batch of flow requests, the corresponding hybrid (wireless and/or wired) routing paths. To do so, we proceed as follows. First, each node/edge in the wireless connectivity graph of HDCN is expanded making use of an advanced Edmonds-Szeider [84] approach. Second, we put forward a new Integer Linear Programming (ILP) formulation of `JBRC` in the expanded graph. It specifically considers both inter-flow and intra-flow interferences while ensuring unsplittable paths. To do so, `JBRC` bears an optimization objective of minimizing the end-to-end delay over all the links of the hybrid routing paths. Finally, to solve large instances of `JBRC`, we propose, first a **heuristic** based solution `JBH-HDCN`, based on $A^{\star}$ search algorithm. Then, we propound an **approximate** solution `SJB-HDCN` based on the Lagrangian relaxation technique [85], to guarantee a lower bound of the optimal solution. Note that our proposals ensures that the obtained routing paths are optimized, unsplittable and free of intra-flow interferences. Based on extensive network simulations conducted in QualNet simulator dealing with the full protocol stack, we assess the performance of our proposal compared to the most relevant related strategies. We consider different traffic patterns: i) uniform traffic pattern based on Poisson distribution, and ii) Facebook DCN traffic workload [81].

The remainder of this chapter proceeds as follows. In Section 6.2, we will present our HDCN model and formulate the joint batch routing and channel assignment problem. Section 6.3 will describe the proposed heuristic-based solution. Besides, we will present our scalable approximate proposal `SJB-HDCN` in Section 6.4. Afterwards, simulation environment and results will be presented in Section 6.5. Finally, we will conclude this work in Section 6.6.

6.2 Problem Formulation

In this section, we will, first, define the model of inter-rack wireless/wired network. Afterwards, we will formulate the joint batch routing and channel assignment problem in HDCN based on an advanced Multi-Commodity Flow (MCF) model.

6.2.1 Hybrid Data Center Network Model

We define a Wireless/Wired Transmission Unit (WTU), denoted by W_i, as a group of servers in a rack sharing a set of wireless beamforming antennas and a gigabit wired switch. Each W_i is equipped with 4 IEEE 802.11ad transceivers/antennas (i.e., orthogonal channels) denoted by $\{w_i^1, w_i^2, w_i^3, w_i^4\}$ and one Top of Rack switch (ToR) based on IEEE 802.3 denoted by w_i^5. Note that the communications between $\{W_i\}$ (i.e., inter-rack) are ensured by both: i) a wireless infrastructure (through $\{w_i^1, w_i^2, w_i^3, w_i^4\}$) and/or ii) a wired infrastructure (through w_i^5).

We model the HDCN as an undirected graph $\mathcal{G} = (V(\mathcal{G}), E(\mathcal{G}))$. Each node $v_i \in V(\mathcal{G})$ corresponds to one WTU W_i. An edge $e \in E(\mathcal{G})$ between two nodes v_i and v_j exists if and only if they can communicate in full-duplex among all the wireless channels of IEEE 802.11ad while assuming the absence of interferences. As in our previous contribution [86], we make use of the Friis signal transmission model. Formally, we model the interference between two communication links $e = (w_i^k, w_j^k)$ and $e' = (w_m^k, w_n^k)$ as follows: i) transmitter, w_i^k of e interferes with receiver w_n^k of e' or ii) transmitter w_m^k of e' interferes with receiver w_j^k of e.

We distinguish two kinds of interferences in HDCN: i) intra-flow and ii) inter-flow interferences. Intra-flow interferences are caused by two successive links belonging to the same path and simultaneously using an identical wireless channel. Thanks to the beamforming technique, intra-flow interference between the non-successive links is avoided. Inter-flow interferences are caused by active links belonging to different paths and transmitting over the same wireless channel. In order to avoid the intra-flow interferences, WTU cannot receive and transmit simultaneously on the same channel. On the other hand, inter-flows interferences are minimized by selecting wireless links with minimal cost in term of retransmission delay.

6.2.2 Joint Batch Routing $\&$ Channel Assignment (JBRC) problem

We model the arrival rate of flow commodities with a Poisson process with an arrival rate λ_A. It is worth noting that in the batch strategy, the communications arriving during a specific time window, denoted δ_T, are queued together and their processing is delayed to the following time window. By the end of δ_T, the joint batch routing and channel assignment procedure is triggered in order to find the adequate routing paths for the incoming communication flows. Consider a set of ζ communication flows (i.e., commodities), arriving during a slot δ_T, $\mathcal{B} = \{(W_{s,i}, W_{d,i}), r_i\}$, $i \in \{1, ..., \zeta\}$, where $W_{s,i}$, $W_{d,i}$ and r_i denote respectively the source WTU, the destination WTU

and the requested flow of the i^{th} communication.
The main reason behind the use of batch arrival model is to enhance HDCN performance. Indeed, the arrival order closely impacts resource allocation as well as the routing paths.

The objective of the joint batch routing and channel assignment problem in HDCN consists in computing, for each δ_T, the set of ζ hybrid (wireless and/or wired) routing paths for all the ζ incoming communications, $\mathcal{B} = \{(W_{s,i}, W_{d,i}), r_i\}$, $i \in \{1, ..., \zeta\}$, in a way that maximizes the total throughput. To do so, we aim to minimize the end-to-end delay by considering i) residual traffic in IP queues of the paths (waiting delay), ii) data rate of network interfaces (transmission velocity), and iii) wireless interferences (retransmission delay).
Therefore, the ζ hybrid routing paths should satisfy: i) elimination of intra-flow interferences by adequately assigning the wireless channels to all successive hops, ii) minimization of inter-flows interferences by minimizing the retransmission cost, iii) minimization of waiting delay by considering both the incoming and residual traffic in the IP queues along the path (wired and/or wireless).

Note that the above routing paths are hybrid (wireless and/or wired). Unfortunately, the undirected weighted graph $\mathcal{G} = (V(\mathcal{G}), E(\mathcal{G}))$ does not include i) wired links and ii) wireless channel links. In fact, an edge in $\mathcal{G}$ between W_i and W_j is the fusion of the four wireless channel links. For this reason, we propose to extend $\mathcal{G}$ to include the missing links. To do that, we adapt a specific node/edge expansion approach inspired from Edmond's Szeider technique [84] to generate a new graph denoted by $\hat{\mathcal{G}} = \left(\hat{V}\left(\hat{\mathcal{G}}\right), \hat{E}\left(\hat{\mathcal{G}}\right)\right)$. $\hat{\mathcal{G}}$ supports simultaneously wired and multi-channel wireless links. The problem of finding the hybrid paths for the set of incoming communications during each window δ_T is formulated as Multi-Commodity Flow (MCF) problem in $\hat{\mathcal{G}}$ as detailed in the next sub-section 6.2.2.1. In fact, the path is built by concatenating the links that transmit flow in $\hat{\mathcal{G}}$. Recall that each edge in $\hat{\mathcal{G}}$ is associated to exactly one interface (wireless channel or wired).

6.2.2.1 Graph Expansion

We revoke that each wireless/wired transmission unit W_i is equipped with 4 wireless interfaces denoted by $\{w_i^1, w_i^2, w_i^3, w_i^4\}$ and the wired ToR switch interface denoted by w_i^5. We transform the graph G to the new expanded graph $\hat{\mathcal{G}}$. The latter is generated using the following operations:

1. Each node $v_i \in V(\mathcal{G})$ corresponding to one WTU W_i is expanded into 5 sub-nodes as follows:

   - 4 wireless sub-nodes referring to the wireless channels: $\{\hat{v}_i^1, \hat{v}_i^2, \hat{v}_i^3, \hat{v}_i^4\}$.
   - 1 wired sub-node $\{\hat{v}_i^5\}$.

   Let $\hat{V_S}\left(\hat{\mathcal{G}}\right)$ denote the set of sub-nodes in $\hat{\mathcal{G}}$.

2. Each pair of sub-nodes $(\hat{v}_i^k, \hat{v}_i^l)$, $k \neq l, k, l \in \{1, .., 5\}$, is attached with zero-cost **internal** link as illustrated in Figure 6.1. We refer to the set of internal links by $\hat{E_I}\left(\hat{\mathcal{G}}\right)$.

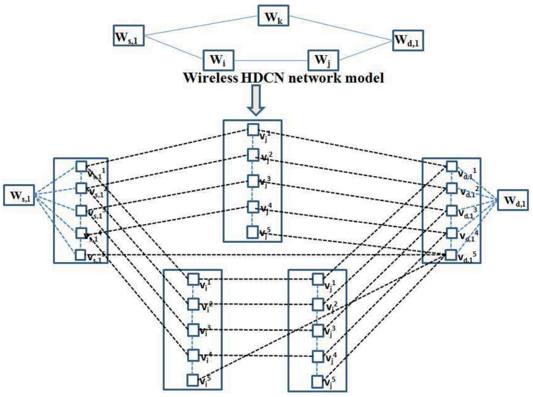


Figure 6.1: Example of graph expansion

3. Each edge $e_i \in E(\mathcal{G})$ is expanded into 4 (i.e., number of channels) **exterior** links denoted by $\{\hat{e}_i^1, \hat{e}_i^2, \hat{e}_i^3, \hat{e}_i^4\}$. If e_i is attaching v_m and v_n that implies each exterior link $\hat{e}_i^k$, $k \in \{1, .., 4\}$ will attach $\hat{v}_m^k$ and $\hat{v}_n^k$ (analogous sub-nodes in term of wireless channel) as shown in Figure 6.1. We refer to the set of external links by $\hat{E_E}\left(\hat{\mathcal{G}}\right)$.

4. We add to $\hat{\mathcal{G}}$ all the distinct sources $W_{s,i}$, and destinations $W_{d,i}$, $i \in \{1, .., \zeta\}$. We denote by $\hat{V_{S'}}\left(\hat{\mathcal{G}}\right)$ the set of these nodes. Afterwards, $\hat{\mathcal{G}}$ is augmented by connecting each $W_{s,i}$ and $W_{d,i}$, $i \in \{1, .., \zeta\}$, node to its corresponding expanded sub-nodes in $\hat{\mathcal{G}}$.

5. Finally, each wired sub-node in $\hat{\mathcal{G}}$ is directly attached to all the destination nodes $W_{d,j}$, $j \in \{1, .., \zeta\}$ through an exterior wired edge (i.e., two-hop wired OSPF path).

Note that $\hat{\mathcal{G}}$ is **weighted** undirected graph where the cost of each **exterior** link $\hat{e} \in \hat{E_E}\left(\hat{\mathcal{G}}\right)$ is equal to:

$$\mathcal{C}(\hat{e}) = \frac{1}{\mathcal{D}(\hat{e})} \cdot [1 + \alpha \cdot \mathcal{F}(\hat{e}) + \frac{\varrho \sum_{\bar{e} \in \mathcal{I}(\hat{e})} \mathcal{R}(\bar{e})}{\alpha}] \tag{6.2.1}$$

where iii) $\mathcal{D}(\hat{e})$ is the data rate of the $\hat{e}$'s sending extremity, ii) $\mathcal{I}(\hat{e})$ is the set of all **active** wireless interfering links with $\hat{e}$, iii) $\alpha = max\{1, |\mathcal{I}(\hat{e})|\}$ is a coefficient reflecting the number of interfering links if they exist, iv) $\mathcal{F}(\hat{e})$ represents the sum of residual and requested traffic volumes in IP queue, v) $\mathcal{R}(\hat{e})$ denotes the residual traffic in the IP queue of an interfering link $\hat{e}$, and vi) ϱ is the maximum number of frame retransmissions and fixed by IEEE 802.11ad standard to 7. We assume that if $\hat{e}$ is wired interface then $|\mathcal{I}(\hat{e})| = 0$. It is worth pointing out that the cost of a link incarnates the transmission delay of its residual traffic and the resulting re-transmission delays caused by/on interfering links.

6.2.2.2 Multi-Commodity Flow problem (MCF) formulation

We formulate the joint batch routing and channel assignment in HDCN as a Multi-Commodity Flow problem in the expanded graph $\hat{\mathcal{G}}$. The latter is defined as a network flow problem formed by multiple commodities. Note that commodities represent in our formulation the flow demands, defined by the : i) source WTU, the ii) the destination WTU and iii) the requested traffic to be transmitted. It is worth pointing out that finding the set of ζ routing paths for the batch $\mathcal{B}$ of communications, in a wired DCN, is equivalent to resolving the multi-commodity flow problem [87] on the graph $\hat{\mathcal{G}}$. In the present work, we deal with joint routing and channel allocation problem in HDCN. Consequently, we seek for the hybrid (wireless and/or wired) routing paths for the different flow commodities. To do so, we propose a new linear formalization of the MCF problem presented hereafter.

We define the flow allocation variable $f^i(\hat{e}) : \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right)$, that indicates the quantity of traffic to be allocated on link $\hat{e}$ for the i^{th} communication. Let $\mathcal{C}(\hat{e})$ denote the cost value of the exterior link $\hat{e} \in \hat{E_E}\left(\hat{\mathcal{G}}\right)$, given by equation 6.2.1. Interior edges are ignored and assigned zero costs because they do not induce interference. In our formulation, our aim consists in allocating the links with minimal costs. In doing so, we minimize the end-to-end delay by considering i) residual traffic in IP queues of the paths (waiting delay), ii) data rate of network interfaces (transmission velocity), and iii) wireless interferences of inter flows (retransmission delay). Note that minimizing the end-to-end delay is, to some extent, equivalent to maximizing the throughput in the HDCN. Formally, the objective function of our problem is described by the equation below:

$$\text{minimize } \mathcal{R} = \textstyle\sum_{\hat{e} \in \hat{E_E}\left(\hat{\mathcal{G}}\right)} \sum_{i=1}^{\zeta} \mathcal{C}(\hat{e}) \cdot f^i(\hat{e}) \tag{6.2.2}$$

Note that the flow allocation variable $f^i(\hat{e})$ is integer and should verify the following constraint:

$$f^i(\hat{e}) \geq 0, \ \ \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right), \forall i \in \{1, .., \zeta\} \tag{6.2.3}$$

We refer by $\hat{E}_{\hat{v}}^{out}\left(\hat{\mathcal{G}}\right)$ and $\hat{E}_{\hat{v}}^{in}\left(\hat{\mathcal{G}}\right)$ to respectively the sets of the outgoing and incoming edges of the node $\hat{v}$ in $\hat{\mathcal{G}}$. The multi-commodity flow problem formulation computes the routing path for each flow i between $W_{s,i}$ and $W_{d,i}$ by guaranteeing the flow conservation constraint given hereafter:

$$\begin{gathered} \textstyle\sum_{\hat{e} \in \hat{E}_{\hat{v}}^{out}\left(\hat{\mathcal{G}}\right)} f^i(\hat{e}) - \sum_{\hat{e} \in \hat{E}_{\hat{v}}^{in}\left(\hat{\mathcal{G}}\right)} f^i(\hat{e}) = 0, \\ \forall \hat{v} \in \hat{V}\left(\hat{\mathcal{G}}\right) \setminus \{W_{s,i}, W_{d,i}\}, \ \forall i \in \{1, .., \zeta\} \end{gathered} \tag{6.2.4}$$

The bandwidth requirement constraint guarantees that the total requested flow r_i is successfully transmitted for each commodity as formulated hereafter:

$$\begin{aligned}&\textstyle\sum_{\hat{e}\in\hat{E}^{out}_{W_{s,i}}(\hat{\mathcal{G}})} f^i(\hat{e}) - \sum_{\hat{e}\in\hat{E}^{in}_{W_{s,i}}(\hat{\mathcal{G}})} f^i(\hat{e}) = r_i,\\ &\forall i \in \{1,..,\zeta\}\end{aligned} \tag{6.2.5}$$

$$\begin{aligned}&\textstyle\sum_{\hat{e}\in\hat{E}^{in}_{W_{d,i}}(\hat{\mathcal{G}})} f^i(\hat{e}) - \sum_{\hat{e}\in\hat{E}^{out}_{W_{d,i}}(\hat{\mathcal{G}})} f^i(\hat{e}) = r_i,\\ &\forall i \in \{1,..,\zeta\}\end{aligned} \tag{6.2.6}$$

It is worth noting that by considering only the above constraints, multi-commodity flow problem may result in path splitting by allocating the same flow on multiple routing paths. In this present work, each flow is transmitted using a single route in order to avoid the costs induced by multi-path routing. Therefore, each edge in the graph can either transmit the full traffic of a communication $i \in \{1,..,\zeta\}$ or none. To do so, we denote by $y^i(\hat{e}) \in \{0,1\}$ a binary variable indicating whether the link $\hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right)$ is transmitting traffic or not for the i^{th} communication. Single path routing is hence expressed by:

$$f^i(\hat{e}) = y^i(\hat{e}) \cdot r^i,\ \ \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right), \forall i \in \{1,..,\zeta\}, y^i(\hat{e}) \in \{0,1\} \tag{6.2.7}$$

To further avoid intra-flow interference (i.e., each wireless node is prohibited from transmitting and receiving simultaneously on the same channel), we enforce each wireless sub-node to: i) participate in at most one flow communication at the same time, and ii) receive and send data on different channels. If $\hat{V}_s\left(\hat{\mathcal{G}}\right)$ denotes the set of wireless sub-nodes in the graph $\hat{\mathcal{G}}$, then this condition is given by the following constraints:

$$\textstyle\sum_{\hat{e}\in\hat{E}^{out}_{\hat{v}}(\hat{\mathcal{G}})\cup\hat{E}^{in}_{\hat{v}}(\hat{\mathcal{G}})} \sum_{i=1}^{\zeta} y^i(\hat{e}) \leq 1,\ \ \forall \hat{v} \in \hat{V}_s\left(\hat{\mathcal{G}}\right) \tag{6.2.8}$$

$$\textstyle\sum_{k=1}^{\zeta} \sum_{\hat{e}\in\hat{E}^{out}_{W_{s,i}}(\hat{\mathcal{G}})} y^k(\hat{e}) = 0,\ \forall i \in \{1,..,\zeta\},\ k \neq i \tag{6.2.9}$$

$$\textstyle\sum_{k=1}^{\zeta} \sum_{\hat{e}\in\hat{E}^{in}_{W_{d,i}}(\hat{\mathcal{G}})} y^k(\hat{e}) = 0,\ \forall i \in \{1,..,\zeta\},\ k \neq i \tag{6.2.10}$$

Note that equations 6.2.9 and 6.2.10 deal with the case when a WTU is a common source or destination of many requests in the same batch.
Moreover, To minimize waiting delay in IP queues of wired nodes, each wired node is prohibited from transmitting or receiving simultaneously for many flows. If $\hat{V}_d\left(\hat{\mathcal{G}}\right)$ denotes the set of wired sub-nodes in the graph $\hat{\mathcal{G}}$, then this condition is given by the following constraints:

$$\textstyle\sum_{\hat{e}\in\hat{E}^{out}_{\hat{v}}(\hat{\mathcal{G}})} \sum_{i=1}^{\zeta} y^i(\hat{e}) \leq 1,\ \ \forall \hat{v} \in \hat{V}_d\left(\hat{\mathcal{G}}\right) \tag{6.2.11}$$

$$\textstyle\sum_{\hat{e}\in\hat{E}^{in}_{\hat{v}}(\hat{\mathcal{G}})} \sum_{i=1}^{\zeta} y^i(\hat{e}) \leq 1,\ \ \forall \hat{v} \in \hat{V}_d\left(\hat{\mathcal{G}}\right) \tag{6.2.12}$$

Problem 4 summaries the formulation of the Joint Batch Routing and Channel assignment problem (JBRC) in HDCN.

$$\begin{array}{ll}
\texttt{minimize} & \mathcal{R} = \sum_{\hat{e}\in \hat{E_E}(\hat{\mathcal{G}})} \sum_{i=1}^{\zeta} \mathcal{C}(\hat{e}) \cdot f^i(\hat{e}) \\
\texttt{subject to:} & \\
 & 6.2.3, 6.2.4, 6.2.5, 6.2.6, 6.2.7, 6.2.8 \\
 & 6.2.9, 6.2.10, 6.2.11, 6.2.12 \\
 & f^i(\hat{e}) : integer, \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right), \forall i \in \{1, .., \zeta\} \\
 & y^i(\hat{e}) : binary, \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right), \forall i \in \{1, .., \zeta\}
\end{array}$$

Problem 4: Formulation of JBRC

It is clear that JBRC is integer linear programming problem since y_e^i and $f^i(\hat{e})$ are integer while $\mathcal{R}$ is linear.

6.3 Heuristic solution: JBH−HDCN

It is worth pointing out that JBRC problem is an advanced formulation of multicommodity flow model, which is in general very hard to solve, due to scalability constraints. In fact, the dimension of the solution space would heavily increase following: i) the number of requests incoming during the time window δ_T of the batch, and ii) the size of the network topology. Unfortunately, the classical Branch&Cut algorithm struggles to scale with large instances. To get rid of the complexity challenge, we propose a new batch joint routing and channel assignment heuristic in HDCN, named JBH−HDCN. In fact, the order of routing the incoming flow requests deeply impacts the efficiency of the wireless resources allocation, and hence, the network performance. Therefore, instead of tackling the whole ILP JBRC problem, our heuristic solution JBH−HDCN processes, for each δ_T, the **best ordered sequence** of the requests in the batch, denoted $\phi_b \in \mathcal{B}$. Specifically, ϕ_b defines the order for which communications are sequentially processed while minimizing the delay (i.e., enhancing the throughput).

Formally, the objective of JBH−HDCN, is to generate the best sequence $\phi_b \in \mathcal{B}$, while: $\forall \phi_i \in \mathcal{B}$, $\mathcal{D}(\phi_i) \leq \mathcal{D}(\phi_b)$, where $\mathcal{D}(\phi_i)$ corresponds to the sum of all the transmission and re-transmission delays induced by the routing of all the communications in the sequence ϕ_i. Note that:

$$\mathcal{D}(\phi_i) = \sum_{c_i \in \phi_i} \sum_{e \in \mathcal{R}_i} \mathcal{C}(e) \tag{6.3.13}$$

where $\mathcal{R}_i$ denotes the routing hybrid path of the communication $c_i \in \phi_i$ and $\mathcal{C}(e)$ represents the cost of link e computed by equation 6.2.1. JBH−HDCN makes use of i) $A^\star$ **search** heuristic, to find the best sequence ϕ_b, and ii) an advanced **Dijkstra** algorithm to jointly route and assign channels.

JBH−HDCN proceeds in three main stages: i) **Initialization**, ii) **Evaluation and selection**, and iii) **Expansion** stages.

6.3.1 Initialization stage

All incoming communication requests, i.e., the start nodes, are queued in a specific list, named `OPEN`. Then, a second empty list `CLOSED`, used for expanded nodes, is initialized.

6.3.2 Cost evaluation and selection stage

`JBH-HDCN` evaluates the expected estimated cost required to reach ϕ_b from each un-expanded node in `OPEN`. The node with minimum cost is selected and added to `CLOSED`. The cost, $f(n)$, of each node n represents the total **estimated** transmission and re-transmission delays along the hybrid paths of communications in the sequence going through n. Formally:

$$f(n) \ = \ \mathcal{D}(\phi_n) \ + \ \textstyle\sum_{m \in \mathcal{B} \setminus \phi_n} \mathcal{P}(n, m) \cdot \mathcal{D}(\phi_m) \tag{6.3.14}$$

where ϕ_n is the sequence between the start and current nodes. The second term represents a heuristic estimate cost of the best path between n and the last node of ϕ_b. It is computed as in [88], where ϕ_i is the sequence going through n, and $\mathcal{P}(n, m)$ denotes the probability to transit to node m from n. We consider equals probabilities for all transitions. To evaluate $\mathcal{D}(\phi_n)$, `JBH-HDCN` computes the routing path and channel assignment of the communication n, in $\hat{\mathcal{G}}$, using an advanced Dijkstra algorithm. Note that the latter computes the shortest path between the source and destination WTUs of the flow n, while allocating channels along the path. To do so, `JBH-HDCN` only selects the non-adjacent exterior links in $\hat{\mathcal{G}}$, so that the intra-flow interference is prohibited. Moreover, we propose to generate the shortest path according to the link cost value given by equation 6.2.1, which takes into account the transmission and re-transmission delay. Indeed, we choose the shortest path offering the lowest network delay. Accordingly, we define the distance of every single path, $\mathcal{P}$, as follows: $d(\mathcal{P}) \ = \ \sum_{e \in \mathcal{P}} \mathcal{C}(e)$. Once the shortest path is found, `JBH-HDCN` updates the edge costs in $\hat{\mathcal{G}}$ and eliminates all the wireless links allocated to n.

6.3.3 Expansion stage

Our solution expands each selected node by generating all its successors (i.e., node in the batch $\mathcal{B} \setminus \{\phi_i\}$). If only one successor is found, then the latter is a goal node, and the best sequence is obtained by tracing the path from the goal back to s. Otherwise, for each successor m, `JBH-HDCN` evaluates its estimated cost, and decides whether it will be expanded.

`JBH-HDCN` repetitively performs the previous stages, until `OPEN` is empty, in which case, the best solution sequence, ϕ_b, is obtained. Note that ϕ_b resides in `CLOSED`. `JBH-HDCN` is summarized in the pseudo Algorithm 12.

Algorithm 12: JBH-HDCN pseudo-algorithm

```
1:  Inputs: $\hat{\mathcal{G}_2} = \left(\hat{V}\left(\hat{\mathcal{G}_2}\right), \hat{E}\left(\hat{\mathcal{G}_2}\right)\right)$, JBRC-HDCN, $\mathcal{B}$
2:  Output: $\phi_b$
3:  OPEN ← $\mathcal{B}$, CLOSED ← $\emptyset$
4:  Evaluate-Estimated-Cost-Of-Nodes-In-OPEN($\hat{\mathcal{G}_2}$, OPEN)
5:  repeat
6:      $n$ ← Select-Node-With-Minimum-Cost(OPEN)
7:      CLOSED ← CLOSED ∪ $\{n\}$
8:      for all successor $s$ of $n$ do
9:          f($s$) ← Evaluate-Estimated-Cost(s)
            if $s \in$ OPEN OR $s \in$ CLOSED then
                if $f(s) \leq$ Cost($s$) then
10:                 Cost($s$) ← f($s$), Predecessor($s$) ← $n$
                    if $s \in$ CLOSED then
11:                     OPEN←OPEN ∪$\{s\}$, CLOSED←CLOSED$\setminus\{s\}$
                else
12:                 Discard $s$
            else
13:             Cost($s$) ← f($s$), Predecessor($s$) ← $n$
14:     end for
15:     Go to Step 6
16: until OPEN = $\emptyset$
17: $\phi_b$ ← CLOSED
```

6.4 Approximate solution: SJB-HDCN

Although JBH-HDCN handles the scalability constraint and guarantees a feasible routing in a reasonable time, it may deteriorate the network performance by giving a far-from-optimal solution. To resolve JBRC while simultaneously considering the dimension challenge and guaranteeing a near optimal solution, we propose a new strategy named **Scalable Joint Batch-Routing and Channel Assignment in HDCN** (SJB-HDCN). SJB-HDCN makes use of the **Lagrangian relaxation** technique [85], in order to converge to a feasible solution with a guaranteed precision. The main idea behind our approach SJB-HDCN is to move the constraints that are considered to be computational, in JBRC, to the objective function and penalize them using non-negative coefficients, named Lagrangian multipliers. Note that SJB-HDCN not only decreases the computation time of the resolution, but also measures a lower bound of the optimal solution.

SJB-HDCN proceeds as follows: First, **Relaxation stage** relaxes the hard constraints in JBRC and defines both the Lagrangian relaxation problem and its dual one. Second, **Lagrangian function and Subgradient evaluation stage** evaluates the Lagrangian function and its subgradient. Third,

Lagrangian Update stage, is performed by iteratively updating the Lagrangian multiplier values, and evaluating the corresponding Lagrangian function and its subgradient. `SJB-HDCN` repetitively processes these updates until reaching the best possible solution. Hereafter, we will detail each stage.

6.4.1 Relaxation stage

`SJB-HDCN` relaxes first the explicit "hard" constraints by bringing them to the objective function so that optimizing the problem becomes easier. Note that the hard constraints incarnate those that increase the time complexity of the original problem `JBRC`. It is straightforward to see that the constraints dealing with all the flows at the same time are the most likely to increase computation time. Therefore, our approach relaxes the constraints 6.2.8, 6.2.11 and 6.2.12. To do so, `SJB-HDCN` penalizes the relaxed constraints by assigning a positive coefficient, named `Lagrangian multiplier`, to each one. For that, we introduce the non-negative Lagrangian multiplier vector $\mu \in R^{|\hat{V}_s(\hat{\mathcal{G}}) \cup \hat{V}_d(\hat{\mathcal{G}})|}$ for the wireless and wired sub-nodes.

Formally, based on equation 6.2.7, the Lagrangian relaxation of `JBRC` problem, denoted by `LR-JBRC`, is given in problem 5.

$$
\begin{array}{l}
L(\mu) = \texttt{minimize} \quad L(y,\mu) \\
\texttt{subject to:} \\
\quad y^i(\hat{e}) \in \{0,1\} \;\; \forall \hat{e} \in \hat{E}\left(\hat{\mathcal{G}}\right), \forall i \in \{1,..,\zeta\} \\
\quad \sum_{\hat{e} \in \hat{E}^{out}_{\hat{v}}(\hat{\mathcal{G}})} (y^i(\hat{e}) \cdot r^i) - \sum_{\hat{e} \in \hat{E}^{in}_{\hat{v}}(\hat{\mathcal{G}})} (y^i(\hat{e}) \cdot r^i)) = 0, \\
\qquad\qquad \forall \hat{v} \in \hat{V}\left(\hat{\mathcal{G}}\right) \setminus \{W_{s,i}, W_{d,i}\}, \; \forall i \in \{1,..,\zeta\} \\
\quad \sum_{\hat{e} \in \hat{E}^{out}_{W_{s,i}}(\hat{\mathcal{G}})} (y^i(\hat{e}) \cdot r^i) - \sum_{\hat{e} \in \hat{E}^{in}_{W_{s,i}}(\hat{\mathcal{G}})} (y^i(\hat{e}) \cdot r^i) = r_i, \\
\qquad\qquad \forall i \in \{1,..,\zeta\} \\
\quad \sum_{\hat{e} \in \hat{E}^{in}_{W_{d,i}}(\hat{\mathcal{G}})} (y^i(\hat{e}) \cdot r^i) - \sum_{\hat{e} \in \hat{E}^{out}_{W_{d,i}}(\hat{\mathcal{G}})} (y^i(\hat{e}) \cdot r^i) = r_i, \\
\qquad\qquad \forall i \in \{1,..,\zeta\} \\
\quad \sum_{\hat{e} \in \hat{E}^{out}_{W_{s,i}}(\hat{\mathcal{G}})} y^k(\hat{e}) = 0, \; \forall i,k \in \{1,..,\zeta\}, \; k \neq i \\
\quad \sum_{\hat{e} \in \hat{E}^{in}_{W_{d,i}}(\hat{\mathcal{G}})} y^k(\hat{e}) = 0, \; \forall i,k \in \{1,..,\zeta\}, \; k \neq i
\end{array}
$$

Problem 5: `LR-JBRC`

Note that the objective function of `LR-JBRC`, named the Lagrangian function, is defined as follows:

$$
L(y,\mu) \; = \; \mathcal{R} \; + \; L_1(y,\mu) \; + \; L_2(y,\mu) \; + \; L_3(y,\mu) \tag{6.4.15}
$$

where $\mathcal{R}$ is the objective function of the original problem `JBRC`, y is the solution vector of `JBRC`, $L_1(y,\mu)$, $L_2(y,\mu)$ and $L_3(y,\mu)$ refer respectively to:

$$
L_1(y,\mu) = \; \sum_{i=1}^{\zeta} \sum_{\hat{v} \in \hat{V}_s(\hat{\mathcal{G}})} \mu_{\hat{v}} (\sum_{\hat{e} \in \hat{E}^{out}_{\hat{v}}(\hat{\mathcal{G}}) \cup \hat{E}^{in}_{\hat{v}}(\hat{\mathcal{G}})} y^i(\hat{e}) - 1) \tag{6.4.16}
$$

$$L_2(y,\mu) = \sum_{i=1}^{\zeta}(\sum_{\hat{v}\in\hat{V}_d(\hat{\mathcal{G}})} \mu_{\hat{v}} \sum_{\hat{e}\in\hat{E}_{\hat{v}}^{out}(\hat{\mathcal{G}})} y^i(\hat{e}) - 1) \tag{6.4.17}$$

$$L_3(y,\mu) = \sum_{i=1}^{\zeta}(\sum_{\hat{v}\in\hat{V}_d(\hat{\mathcal{G}})} \mu_{\hat{v}} \sum_{\hat{e}\in\hat{E}_{\hat{v}}^{in}(\hat{\mathcal{G}})} y^i(\hat{e}) - 1) \tag{6.4.18}$$

The Lagrangian multipliers $\mu_{\hat{v}}$ are non-negative coefficients that we interpret as the price of the intra-flow interference for each sub-node $\hat{v}$.
It is worth pointing out that the value of $L(\mu)$, for any μ, is a **lower bound** of the optimal objective function of JBRC. Therefore, in order to enhance HDCN performance, SJB-HDCN aims to get the sharpest possible lower bound that is close to the optimal solution. To do so, our approach associates to LR-JBRC problem its dual, named Lagrangian dual problem, and denoted LD-JBRC, defined in Problem 6.

$$\begin{array}{ll} L^* = \texttt{maximize} & L(\mu) \\ \texttt{subject to:} & \mu \geq 0 \end{array}$$

Problem 6: DL-JBRC

In fact, the optimal solution L^* of LD-JBRC is a lower bound of the optimal solution of the JBRC. With this assumption, the optimal solution vector μ^* of LD-JBRC problem corresponds to the optimal solution of the dual of JBRC problem [71] [89].

6.4.2 Evaluating the Lagrangian function and its subgradient

To solve the LR-JBRC, for each value of μ, our approach SJB-HDCN evaluates the Lagrangian function $L(\mu)$. It is worth noting that, $L(\mu)$ is **concave** since it is the minimum of linear forms in μ. Moreover, it is clear to see that it is **non-differentiable**. Furthermore, it is straightforward to notice that none of the constraints of LR-JBRC contains variables for more than one commodity flow. Therefore, for any value of μ, our approach naturally decomposes LR-JBRC into a set of ζ independent **single commodity flow** problems (i.e., one for each commodity) [90] that can be easily solved. Consequently, the Lagrangian function $L(\mu)$ is obtained, for each μ.

Once $L(\mu)$ is evaluated, SJB-HDCN computes its subgradient. Note that a subgradient of the non-differentiable concave function $L(\mu)$ on μ_1 is defined as the vector $S \in R^{|\hat{E}_E(\hat{\mathcal{G}})|}$ that verifies:

$$L(\mu_1) \ \leq \ L(\mu_2) \ + \ S\cdot(\mu_1 - \mu_2), \ \forall\mu_2 \tag{6.4.19}$$

Accordingly, SJB-HDCN computes the subgradients S_1, S_2 and S_3 of respectively $L_1(\mu)$, $L_2(\mu)$ and $L_3(\mu)$, on μ as follows:

$$S_1 \ = \ \sum_{i=1}^{\zeta}(\sum_{\hat{v}\in\hat{V}_s(\hat{\mathcal{G}})} \sum_{\hat{e}\in\hat{E}_{\hat{v}}^{out}(\hat{\mathcal{G}})\cup\hat{E}_{\hat{v}}^{in}(\hat{\mathcal{G}})} y^i(\hat{e}) - 1) \tag{6.4.20}$$

$$S_2 \ = \ \textstyle\sum_{i=1}^{\zeta}(\sum_{\hat{v}\in\hat{V}_d(\hat{\mathcal{G}})}\sum_{\hat{e}\in\hat{E}_{\hat{v}}^{out}(\hat{\mathcal{G}})} y^i(\hat{e}) - 1) \tag{6.4.21}$$

$$S_3 \ = \ \textstyle\sum_{i=1}^{\zeta}(\sum_{\hat{v}\in\hat{V}_d(\hat{\mathcal{G}})}\sum_{\hat{e}\in\hat{E}_{\hat{v}}^{in}(\hat{\mathcal{G}})} y^i(\hat{e}) - 1) \tag{6.4.22}$$

Consequently, the subgradient S of $L(\mu)$ is $S = S_1 + S_2 + S_3$. It is worth pointing out that the subgradient S can be interpreted as the rate of intra-flow interference among the wireless and wired sub-nodes. In other words, it represents the total exceeding on (wireless/wired) interface use, by many links simultaneously.

With the ability of evaluating the Lagrangian $L(\mu)$ function and its subgradient S on μ, our method SJB-HDCN makes use of the subgradient method rules that repetitively update the Lagrangian multipliers in order to reach the optimal solution L^*. Hereafter, we will detail these update rules.

6.4.3 Lagrangian update stage

SJB-HDCN repetitively updates the Lagrangian multipliers μ until reaching the optimal solution L^*. To do so, it makes use of the subgradient method rules [91], and proceeds in three steps:

6.4.3.1 Initialization

SJB-HDCN sets the initial multiplier value μ^0 to $zero$ and resolves the corresponding LR-HDCN problem. The solution $Y^0 = \{y_e^0, \forall e \in \hat{E}\left(\hat{\mathcal{G}}\right)\}$, the Lagrangian function $L(y, \mu^0)$ for μ^0, and its subgradient S^0 are hence obtained.

6.4.3.2 Update of Lagrangian multipliers

At each iteration q, SJB-HDCN computes the new Lagrangian multiplier $\mu^{(q+1)}$ for the next iteration (i.e., $q + 1$) using the following Lagrangian update formula:

$$\mu^{(q+1)} \ = \ max\{(\mu^{(q)} + \theta^{(q)} \cdot S^q), 0\} \tag{6.4.23}$$

where S^q denotes the subgradient of L at $\mu^{(q)}$, and $\theta^{(q)}$ represents the step size. Note that the latter is a crucial parameter that heavily impacts the convergence speed. In fact, it reflects how far our algorithm SJB-HDCN moves from the current solution to the optimal one. Indeed, at each iteration, SJB-HDCN takes a step in the direction of the optimal solution.

Our approach makes use of the diminishing step size rule, where, $\theta^{(q)}$ satisfies the following convergence conditions [91]:

$$\theta^{(q)} \implies 0, \textstyle\sum_{q=1}^{\infty}\theta^{(q)} \implies \infty \tag{6.4.24}$$

Typically, a scalar value of the step size is: $\theta^{(q)} \ = \ h/\sqrt{q}$, where h is a constant value. Note that, for the diminishing step size rule, our method SJB-HDCN is guaranteed to converge to the optimal solution. More specifically, at each iteration q, $L(\mu^q) \ - \ L^* \ = \ \epsilon$, where ϵ is a function of $\theta^{(q)}$ and decreases with it [91].

Algorithm 13: SJB-HDCN pseudo-algorithm

```
Inputs: $\hat{\mathcal{G}}_2 = \left(\hat{V}\left(\hat{\mathcal{G}}_2\right), \hat{E}\left(\hat{\mathcal{G}}_2\right)\right)$, JBRC
Output: $\mathcal{L}^*$
$\mu^0 \leftarrow$ Initial-Lagrangian-Multiplier($\hat{\mathcal{G}_2}$)
$\mathcal{LR}_0 \leftarrow$ Initial Lagrangian relaxation problem(JBRC)
$q \leftarrow 0$
repeat
    $L(\mu^q) \leftarrow$ Compute-Lagrangian-Function($\mathcal{LR}_q$))
    $\mathcal{S}^q \leftarrow$ Compute-Subgradient($L(\mu^q)$))
    $\mu^{q+1} \leftarrow$ Compute-Multiplier($\mu_q, \mathcal{S}^q$), $q \leftarrow q + 1$
until $\mathcal{S}^q = 0$
$\mathcal{L}^* \leftarrow L(\mu^q)$
```

6.4.3.3 Computation of the current Lagrangian function

After each Lagrangian multiplier update, our approach resolves the new Lagrangian problem for μ^{q+1}. In doing so, it evaluates the current Lagrangian function $L(\mu_q)$ and its subgradient. Then, it comes back to the second step and updates the multiplier value.

This process is repeatedly executed until the subgradient of $L(\mu)$ on μ equals **zero** (i.e., no relaxed constraint is violated), in which case the **optimal** solution of DL-JBRC is obtained.

SJB-HDCN is summarized in the pseudo Algorithm 13.

6.5 Performance Evaluation

In this section, we will report the performance of our batch strategies by performing a series of detailed simulations. We start with describing the stages of our implementation and environment set up. Afterwards, we define the performance metrics we consider to evaluate our strategies. Finally, we discuss the obtained results.

6.5.1 Simulation Environment and Methodologies

6.5.1.1 Experiment Design

We make use of QualNet, an event driven network simulation platform based on C++ language, and widely used by the network research community. We integrate new features to QualNet in order to support next generation Multi-Gbps WiFi. Further details about IEEE 802.11ad implementation can be found in section 4.4.

We set the propagation parameters and rate table based on the IEEE 802.11ad. We assume that all the antennas have the same transmission power which is fixed to 10 dBm. We configure the QualNet physical layer with the free-space propagation model, by setting the Friis parameter α to

2. Rx_Thr and CP_Thr values are respectively set to -78 dBm and 10. Furthermore, 4 wireless channels are available according to IEEE 802.11ad specification, with a bandwidth of 2.16 GHz and running frequencies ranging from 57 GHz to 66 GHz.

To deploy beamforming technique, we associate 4 switched-beam antennas, composed of 8 beams, to each ToR. Besides, we build our large scale data center based on a Cisco's MSDC model, containing 256 racks [22], in which we: i) use OSPF protocol for traffic routing and ii) implement ECMP protocol in order to balance the load over the wired network. Each rack typically contains from 20 to 40 servers and the overall infrastructure includes more than 5000 servers. The geographic dimensions are 60m$\times$60m. Servers of the same rack are interconnected through a leaf switch (i.e., ToR). Each leaf is connected to 4 spine switches. As in [22], ToRs (i.e., leaves) are connected to servers via 1 Gbps links. Moreover, spine and leaf switches communicate through 10 Gbps links. In fact CISCO has found out that using multiple 10 Gbps links between spine and leaf instead of a single 40 Gbps link alleviates power consumption in Clos topology. Indeed, The current power consumption of a 40 Gbps optics is more than 10X a single 10 Gbps. Similarly to [8], we set the propagation delay of wired links to 2 μs. The noise factor and implementation loss values are respectively set to 10, and 5, as it is given by IEEE 802.11ad specification [23].
Finally, we implemented i) our exact solution, `BR-HDCN`, based on B&C algorithm using Cplex solver, ii) our heuristic solution `JBH-HDCN` based on C++ languange, iii) our approximate scalable batch approach `SJB-HDCN` based on C++ languange and Cplex solver, and iv) the related strategies.

6.5.1.2 Simulation setup

Regarding the simulations setup, we run our experiments under different workloads. The traffic follows a Constant Bit Rate (CBR) model for which we set the inter-arrival packet time to 6 μ-seconds and the CBR packet size to 6214 Bytes. Note that the latter value is calibrated in a way that no fragmentation occurs during the encapsulation process. In fact, the maximum size of IEEE 802.11ad frame is 7995 Bytes [23]. We make use of UDP transport protocol to transmit the inter-rack traffic.

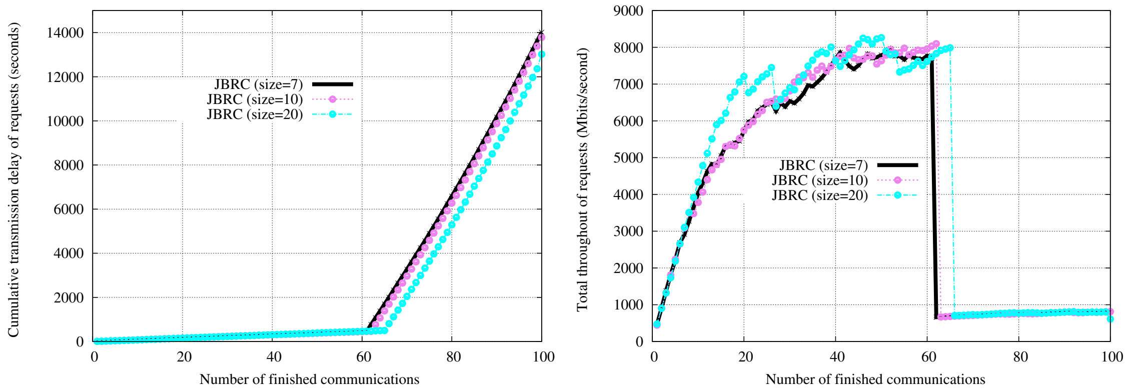
We run the simulation for 100 transmission demands. The confidence interval is fixed to 95%.

6.5.2 Performance metrics

We consider several metrics to evaluate purposes in our experiments:

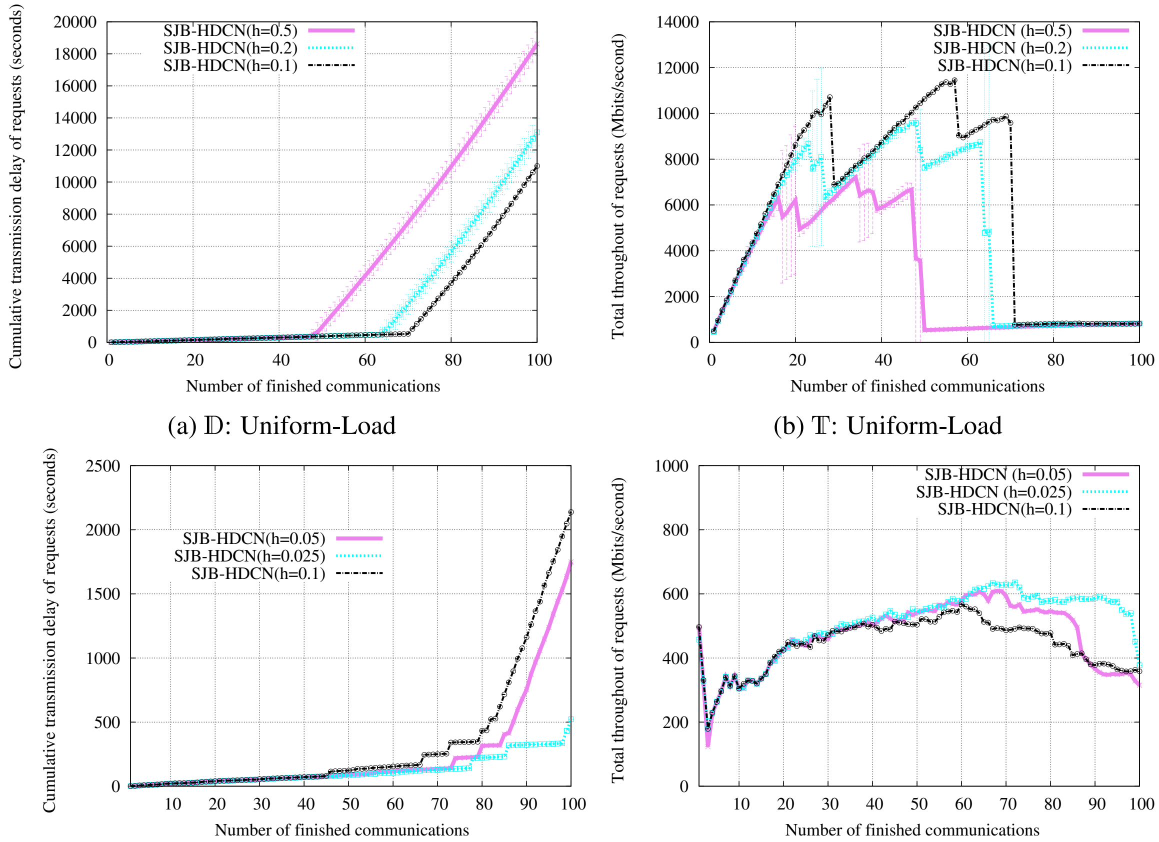
1. $\mathbb{D}$: is the cumulative delay of the network. It defines the cumulative transmission delay of all the finished communications in the network.

2. $\mathbb{D}_{\mathbb{A}}$: is the average delay of the network. It defines the average transmission delay of all the finished communications in the network.



(a) $\mathbb{D}$: Uniform-Load (b) $\mathbb{T}$: Uniform-Load

Figure 6.2: Time window variation: BR−HDCN



(a) $\mathbb{D}$: Uniform-Load (b) $\mathbb{T}$: Uniform-Load

(c) $\mathbb{D}$: Real-Load (d) $\mathbb{T}$: Real-Load

Figure 6.3: Performance of SJB−HDCN for different step size constant h

Table 6.1: Average network metrics: Uniform-Load

	$\mathbb{D}_a$	$\mathbb{T}_a$
`SJB-HDCN`	$24.21 \pm 5.96\%$	$206.11 \pm 32.1\%$
`JRCA-HDCN`	$35.09 \pm 8.25\%$	$151.18 \pm 26.33\%$
`Flyway-HDCN`	$330.79 \pm 2.59\%$	$8.70 \pm 0.063\%$
`Wired-ECMP-HDCN`	$331.39 \pm 2.69\%$	$8.62 \pm 0.12\%$
`Wired-HDCN`	$339.93 \pm 4.73\%$	$8.056 \pm 0.30\%$

3. $\mathbb{T}$: is the total throughput of the network. It corresponds to the cumulative transmission throughput of the traffic carried through the hybrid DCN.

4. $\mathbb{T}_{\mathbb{A}}$: is the average throughput of the network. It corresponds to the average transmission throughput per request of the traffic carried through the hybrid DCN.

5. $\mathbb{R}_L$: is the **R**esidual wire**L**ess traffic. It corresponds to the remaining amount of traffic to be transmitted over the ongoing wireless communications.

6. $\mathbb{R}_D$: is the **R**esidual wire**D** traffic. It corresponds to the remaining amount of traffic to be carried by the ongoing wired communications.

7. $\mathbb{S}_{ia}$: is the average Spatial Spectrum Reuse of the i^{th} channel, $i \in \{1, .., 4\}$.

6.5.3 Simulation Results

To assess the efficiency of our proposals, we consider two main scenarios: i) Uniform-Load scenario, where inter-rack communications arrive independently following a Poisson process, with a uniform flow distribution, and ii) Real-Load scenario, dealing with the recent real workload of **Facebook**'s DC [81].

6.5.3.1 Uniform-Load

In this scenario, we generate inter-rack communication flows whose start time follows a Poisson process, similarly to [41], with an arrival mean λ_A equal to 4 communications per second. The sending WTU is randomly selected using a uniform distribution in the set of racks deployed in the HDCN. Then, the destination WTU is randomly selected by a uniform distribution among the racks that are not in the same transmission range of the sender. The volume of data to transmit for each communication follows a random uniform distribution between 3 and 4 Gbytes.

We proceed as follows. First, we run the exact solution to obtain the optimal solution of the `JBRC` problem for small instances. Second, we run experiments in order to calibrate the **step**

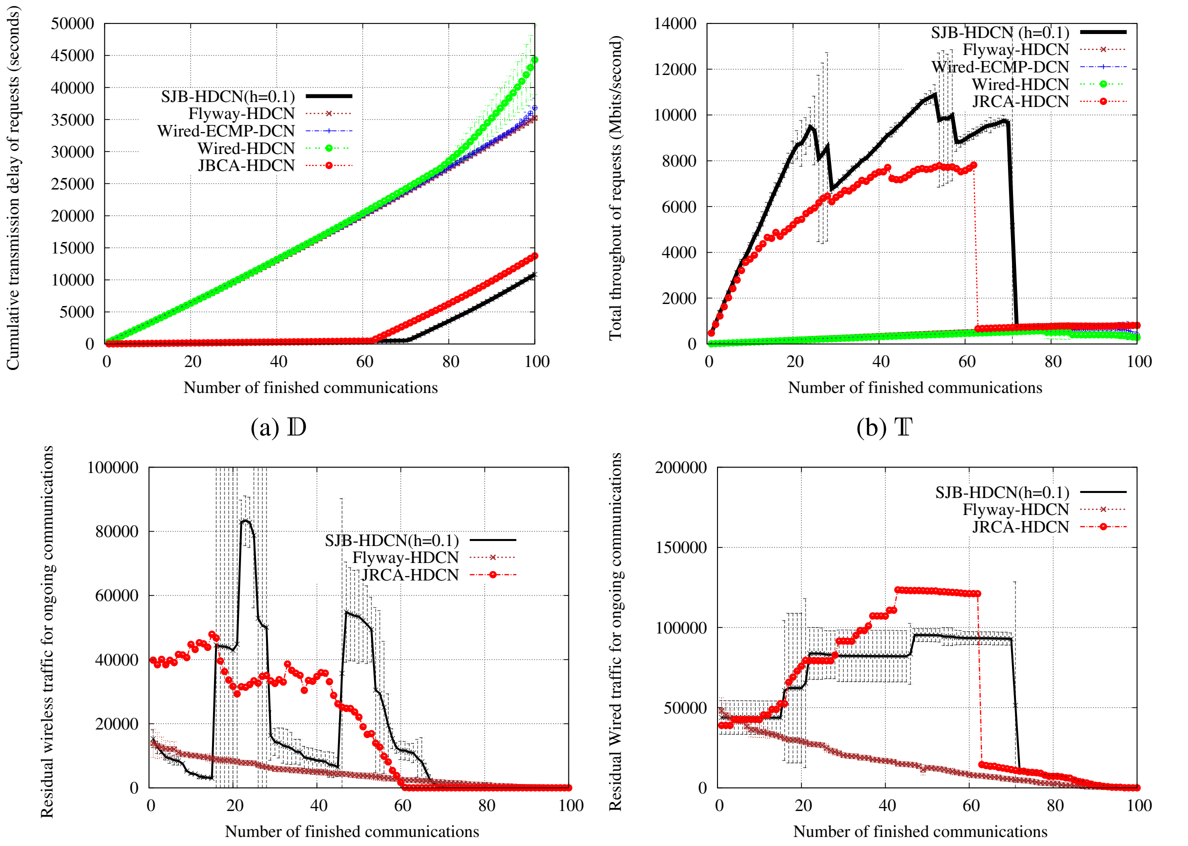
size parameter to the suitable value, $\theta^{(q)}$, of SJB-HDCN. Third, we compare our batch strategy SJB-HDCN to both JRCA-HDCN and related online methods.

Time window variation: We vary the time window δ_T. In fact, δ_T is a decisive parameter since it impacts the size of requests in the batch, and hence the BR-HDCN performance. Figure 6.2(a) and Figure 6.2(b) illustrate network performance of our exact solution BR-HDCN, while varying the batch size. It is clear to see that the larger the batch, the better is the HDCN performance. However, after deep experiments, we noticed that the exact algorithm Branch&Cut, is unable to solve JBRC (CPLEX solver has taken more than 20 hours) when ζ is greater or equal to 40. In the remainder of experiments, we make use of our heuristic and approximate approaches, JBH-HDCN and SJB-HDCN, and we set ζ to 40 for the Uniform-Load scenario.

SJB-HDCN parameter setting Next, we calibrate the step size $\theta^{(q)}$ at each iteration q which is a key parameter of SJB-HDCN since it simultaneously impacts: i) the solution quality, and ii) the iterations number (i.e., the complexity of the algorithm). Therefore, it is very crucial to fix the fastness level of SJB-HDCN while guaranteeing a close-to optimal solution. We run Uniform-Load simulations with a step size $\theta^{(q)} = h/\sqrt{q}$, while varying h in the values: $\{0.5; 0.2; 0.1\}$. We study, through Figure 6.3(a) and Figure 6.3(b), the impact of the step size on both the total network delay and throughput in the HDCN. It is clear to see that the best network performance is ensured for a step size of $0.1/\sqrt{q}$. Accordingly, in the remainder of Uniform-Load simulations, we set h to 0.1.

Comparison with online approaches We first consider the online problem. We compare our proposed online strategy JRCA-HDCN to the related online approaches i) Flyway-HDCN, ii) Wired-ECMP-HDCN and iii) Wired-HDCN. Afterwards, we run our scalable batch strategy SJB-HDCN in order to prove its efficiency towards the online methods.
that routes the set of communications arriving during δ_T, We first evaluate the cumulative delay of the network, $\mathbb{D}$. The results are illustrated in Figure 6.4(a). It is straightforward to see that our batch strategy SJB-HDCN ensures the lowest cumulative delay compared to all the online methods. Besides, our online proposal JRCA-HDCN importantly reduces the delay compared to the related online strategies. Indeed, by the end of communications, SJB-HDCN reduces the total network delay by 19.84% compared to JRCA-HDCN. Such a result proves that the batch processing of communications enhances the HDCN performance. Moreover, JRCA-HDCN reduces $\mathbb{D}$ by 61.21%, 61.93% and 66.94% respectively compared to Flyway-HDCN, Wired-ECMP-HDCN and Wired-HDCN. Table 6.1 illustrates the average transmission delay of the totality of communication demands. We remark that both SJB-HDCN and JRCA-HDCN ensure the lowest average delay.

The total throughput, $\mathbb{T}$, obtained by the considered approaches, is depicted through Figure 6.4(b). This figure clearly shows that SJB-HDCN achieves the highest total throughput com-



(a) $\mathbb{D}$

(b) $\mathbb{T}$

(c) $\mathbb{R}_L$

(d) $\mathbb{R}_D$

Figure 6.4: Uniform-Load scenario

pared to the online strategies, while our online proposal JRCA-HDCN enhances $\mathbb{T}$ compared to the related approaches. In fact, by the end of transmissions, our proposal improves the throughput respectively by 2.35%, 52.12% and 65.81% compared to Flyway-HDCN, Wired-ECMP-HDCN and Wired-HDCN strategies. Note that the total throughput decreases by the end of the simulation. This is due to the late departure of wired communications, which results in high delay and low final throughput. The above results corroborate those obtained for the average network throughput presented in Table 6.1. It is clear to see that our batch strategy SJB-HDCN improves the average throughput by approximately 26.65% compared to our online method JRCA-HDCN. Similarly, the latter enhances $\mathbb{T}_{\mathbb{A}}$ by approximately 94% compared to the three online related approaches.

To study the impact of both batch and online strategies on the wireless resource use, we evaluate, through Table 6.2 the average Spatial Spectrum Reuse $\mathbb{S}_{ia}$ for each channel w^i. We notice that both our batch and online proposals enhance the $\mathbb{S}_{ia}$. In fact, we remark that our methods guarantee a $\mathbb{S}_{ia}$ value much higher than that of Flyway-HDCN method. This weak channel re-utilization

Table 6.2: Average Spectrum Spatial Reuse: Uniform-Load

	SJB-HDCN	JRCA-HDCN	Flyway-HDCN
$\mathbb{S}_{1a}$	14.43 ± 20.06	$16.93 \pm 1.08\%$	$1.08 \pm 0.12\%$
$\mathbb{S}_{2a}$	15.20 ± 26.15	$16.48 \pm 1.07\%$	$1.022 \pm 0.14\%$
$\mathbb{S}_{3a}$	16.28 ± 18.3	$15.47 \pm 1.14\%$	$0.67 \pm 0.14\%$
$\mathbb{S}_{4a}$	13.43 ± 1.22	$15.87 \pm 1.20\%$	$0.55 \pm 0.11\%$

Table 6.3: Average network metrics: Real-Load

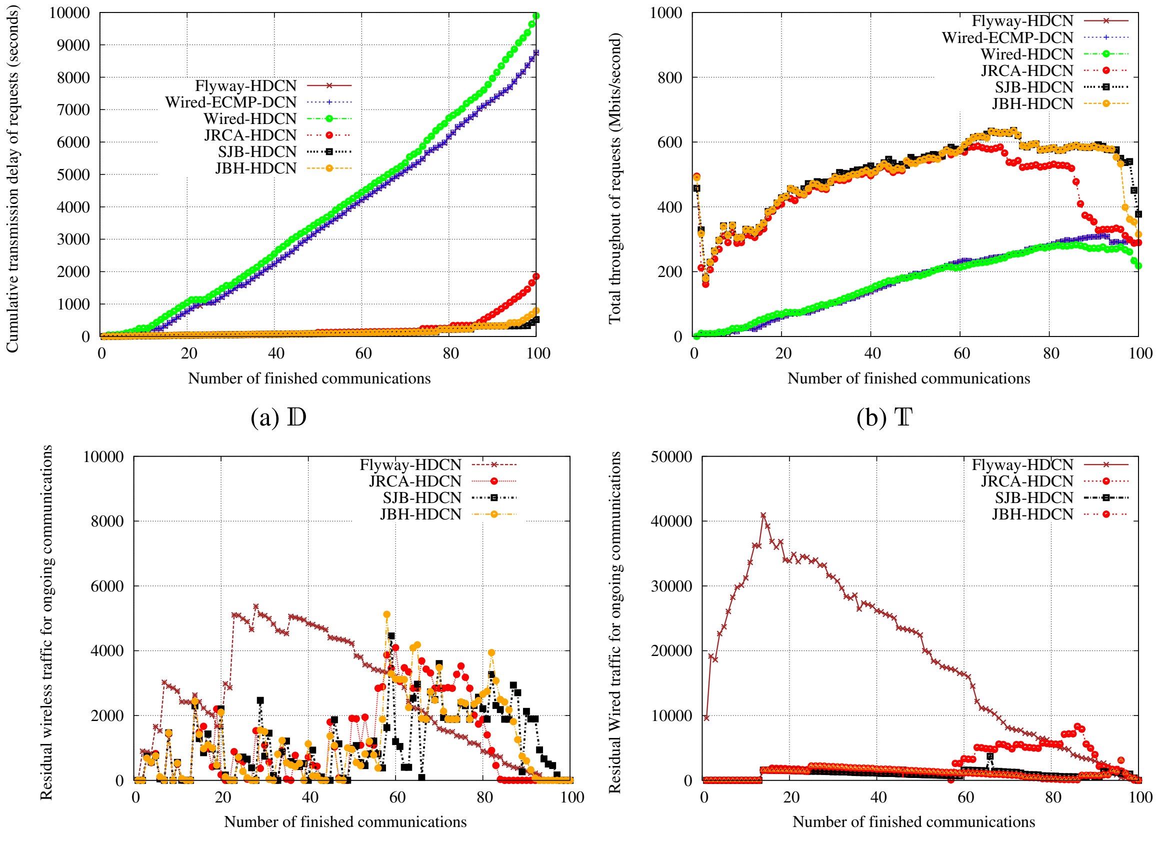
	$\mathbb{D}_a$	$\mathbb{T}_a$
SJB-HDCN	$2.16 \pm 0.54\%$	$20.76 \pm 32.1\%$
JRCA-HDCN	$3.45 \pm 11.84\%$	$19.45 \pm 26.33\%$
JBH-HDCN	$2.29 \pm 1.26\%$	$20.89 \pm 0.12\%$
Flyway-HDCN	$56.91 \pm 35.17\%$	$3.19 \pm 0.52\%$
Wired-ECMP-HDCN	$57.45 \pm 19.15\%$	$3.18 \pm 0.49\%$
Wired-HDCN	$64.52 \pm 4.38\%$	$3.17 \pm 0.11\%$

strongly impacts the performance of the communications as well as the residual wireless and wired resources. In fact, as depicted in Fig 6.4(c) and Fig 6.4(d), the efficient use of the spectrum results in a high residual wireless resources $\mathbb{R}_L$ and low residual wired resources $\mathbb{R}_D$.

6.5.3.2 Real-Load

In this scenario, we consider the flow traces recently generated by **Altoona Facebook**'s data center [81]. In fact, Facebook monitoring system, fbflow, has collected, in 2015 for a period of 24-hours, samples of traffic patterns inside the production clusters. Facebook has made accessible flow workload of some applications, namely: Hadoop, Web-servers, and Database. In our simulations, we consider of the inter-rack traffic generated by Hadoop, since it is considered to be the heaviest [81].

Similarly, we proceed as follows. First, we run experiments in order to calibrate the step size $\theta^{(q)}$ to the suitable value. In fact, the Hadoop's traffic is very unbalanced and varies in a different way compared to the uniform distribution. Consequently, experiments analysis show that the best step size value obtained for the Uniform-Load scenario does not obviously guarantee the best solution for Hadoop workload. Second, we compare our batch strategies to our online algorithm JRCA-HDCN, as well as to the related online strategies.



(a) $\mathbb{D}$

(b) $\mathbb{T}$

(c) $\mathbb{R}_L$

(d) $\mathbb{R}_D$

Figure 6.5: Real-Load scenario: Hadoop cluster in Facebook

SJB−HDCN parameter setting We run simulations for Hadoop traffic while varying the constant h between the values: $\{0.1; 0.05; 0.025\}$. Figure 6.3(c) and Figure 6.3(d) show that the best network performance is ensured for a step size of $0.025/\sqrt{q}$. Accordingly, in the remainder of Real-Load simulations, we set $\theta^{(q)}$ to the best value, i.e., $0.025/\sqrt{q}$.

Comparison between batch and online approaches We consider the online approach JRCA−HDCN, where each single Hadoop flow is routed as it arrives. We compare the performance of JRCA−HDCN to the related online approaches i) Flyway−HDCN, ii) Wired−ECMP−HDCN and iii) Wired−HDCN. Afterwards, we consider the set of communications arriving during a $\delta_T = 2s$. Hadoop workload shows that the traffic is very unbalanced and heavy for most of inter-rack communications, which leads to large sized JBRC problem. Therefore, to deal with scalability challenge, we run both our approximate and heuristic batch strategies SJB−HDCN and JBH−HDCN and compare them to the online methods.

Table 6.4: Average Spectrum Spatial Reuse: Real-Load

	SJB-HDCN	JBH-HDCN	JRCA-HDCN	Flyway-HDCN
$\mathbb{S}_{1a}$	3.56 ± 1.08	4.86 ± 0.65	$4.01 \pm 0.38\%$	$0.27 \pm 0.06\%$
$\mathbb{S}_{2a}$	3.04 ± 0.69	3.85 ± 0.46	$3.34 \pm 0.52\%$	$0.105 \pm 0.01\%$
$\mathbb{S}_{3a}$	3.24 ± 0.57	2.3 ± 1.73	$2.08 \pm 0.52\%$	$0.35 \pm 0.08\%$
$\mathbb{S}_{4a}$	3.31 ± 1.22	1.57 ± 0.73	$2.29 \pm 0.39\%$	$0.75 \pm 0.23\%$

We first evaluate the cumulative delay of the network, $\mathbb{D}$. The results are illustrated in Figure 6.5(a). It is straightforward to see that the batch strategies SJB-HDCN and JBH-HDCN guarantee a low cumulative delay compared to all the online methods. Moreover, our approximate batch solution SJB-HDCN performs better than our heuristic approach JBH-HDCN. Besides, our online proposal JRCA-HDCN importantly reduces the delay compared to the related online strategies. Indeed, by the end of communications, SJB-HDCN reduces the total network delay by 71.81% compared to JRCA-HDCN, while JBH-HDCN alleviates $\mathbb{D}$ by 57.01%. This proves that the batch routing enhances the HDCN performance. Furthermore, JRCA-HDCN drastically reduces delay compared to Flyway-HDCN and Wired-ECMP-HDCN.

These results corroborate those of the average transmission delay, illustrated in Table 6.3. We remark that our batch approaches SJB-HDCN and JBH-HDCN and our online method JRCA-HDCN ensure the lowest average delay compared to the related methods. This decrease in the network delay comes with the benefits of enhancing the throughput. In fact, for our batch, SJB-HDCN and JBH-HDCN, and online, JRC-HDCN methods, $\mathbb{T}$ is roughly two to three times higher than that of the related online strategies.

Furthermore, we evaluate, through Table 6.4 the average Spatial Spectrum Reuse $\mathbb{S}_{ia}$ for each channel w^i. We notice that while SJB-HDCN makes use of all the wireless channels with the same frequency, our proposals in general enhance $\mathbb{S}_{ia}$. Consequently, as depicted in Fig 6.5(c) and Fig 6.5(d) the efficient use of the spectrum results in a high residual wireless resources $\mathbb{R}_L$ and low $\mathbb{R}_D$. Note, however, that Flyway-HDCN shows a higher residual wireless traffic at the beginning, due to the waiting delay incured by wired switches. This proves that the creation of flyways is not enough to alleviate the congestion of Facebook's DC, caused by the heavy Hadoop traffic.

6.6 Conclusion

In this chapter, we addressed the problem of joint batch-routing and wireless channel assignment in hybrid data center networks. To alleviate congestion effects, we resort to augmenting the wired DCN with wireless links (IEEE 802.11ad standard) while minimizing interferences (60 GHz 2D beamforming technique). We formulated our problem as an advanced multicommodity flow mode considering both intra-flow and inter-flow interference constraints while prohibiting path splitting. We bear the scalability challenge of the problem by proposing two new scalable approaches: i) a heuristic solution, based on the $A^\star$ search algorithm minimizing JBRC complexity and ii) an approximate solution, using the Lagrangian relaxation technique to reduce computation time and measures a lower bound of the optimal solution. Extensive simulations conducted within QualNet simulator, for both uniform and real Facebook workload, show that our approach outperforms the most related strategies for all network metrics.

Chapter 7

Conclusions

Contents

7.1 Introduction

In this chapter, we will conclude the thesis and provide a glimpse of our future work. In section 7.2, we will summarize the propounded proposals of this thesis. Next, in section 7.3, we will discuss the future research directions that we will focus on, in short and long term views, so as to improve our proposals. Finally, we will summarize, in section 7.4, the list of publications that we have accomplished in this thesis.

7.2 Summary of contributions

In this thesis, we addressed the problem of routing and wireless resource allocation in hybrid (wireless/wired) data center networks. Specifically, our main focus is to deal with the oversubscription problem in traditional wired data center network architectures. To do so, we resort to augmenting the wired infrastructure with inter-rack wireless links so that to alleviate congestion level on switches. In fact, motivated by the feasibility of the new emerging 60 GHz technology and its high offered data rate ($\approx$ 7 Gbps), we envision, a hybrid (wireless/wired) DCN architecture based on i) Cisco's Massively Scalable Data Center (MSDC) model and ii) IEEE 802.11ad standard.

A main challenge of our research is to afford optimal routing and wireless resource allocation strategies for intra-DCN communication flows, while alleviating the congestion of wired infrastructure, and enhancing the network performance. The key insight of such a problem is to harness both wireless and wired interfaces to improve the data center network capabilities in term of bandwidth. To do so, wireless channels have to be properly assigned in such a way that maximizes the amount of traffic transiting over the wireless infrastructure, while mitigating interference effects.

The above problem has been proven NP-hard [8] due to interference constraints and the limited number of available channels in HDCN. Therefore, we get rid of this complexity by tackling the issue in three separate stages. In the first stage, we addressed the wireless channel allocation problem in HDCN in order to find the efficient channel assignment scheme for single-hop communications, by assuming that the communicating racks are placed in the same wireless transmission range. In the second stage, we propounded a new online joint routing and wireless channel assignment mechanism that sequentially computes the optimal hybrid (wireless/wired) routing path for each multi-hop communication in an online mode. Finally, in the third stage, we handled the batched arriving of multi-hop inter-rack communications to the data center. Both a heuristic-based approach and an approximate solution are proposed to solve this problem.

Hereafter, we will summarize our main contributions.

The first contribution is a survey of data center network architectures. Mainly, the existing DCN designs are classified into three groups. The first group includes, switch-centric DCN architectures, which are exclusively wired and hierarchic. The second group consists of server-centric DCN structures that are recursively designed and where servers are enhanced to handle routing functions. The third group comprises the enhanced DCN architectures deploying either optical or wireless technologies in order to overcome the congestion problem in wired infrastructure.

The second contribution is an in-depth overview of the routing and channel allocation strategies in HDCN. The related approaches are classified into three main classes. The first class regroups the strategies dealing with one-hop inter-rack communications in HDCN and proposing wireless channel allocation algorithms to enhance DCN performance. On the other hand, the second class includes the strategies tackling the problem of joint routing and channel assignment in HDCN to process each single multi-hop communication in an online mode. Finally, the third class deals with the joint batch routing and channel assignment problem in HDCN. Only few methods are proposed, so far, in this context to handle the batched arrival of flow requests.

The third contribution addresses the problem of wireless channel allocation of one-hop inter-rack communications in HDCN. The main objective is to maximize the total throughput by maximizing the proportion of communications transiting over the wireless infrastructure while prohibiting interferences. In doing so, both the end-to-end delay in the HDCN and the congestion on wired switches are minimized. The problem is formulated as minimum graph coloring which is NP-Hard. The proposed approach, wireless channel allocation in HDCN based on Graph Coloring,

GC-HDCN, makes use of i) column generation and ii) branch and price optimization schemes. Simulations results show that the proposed solution outperforms most of the relevant related strategies.

As a fourth contribution, we propose a new advanced strategy, named **Joint Routing and Channel Allocation in HDCN** (**JRCA-HDCN**), to handle multi-hop inter-rack communications. Our online approach JRCA-HDCN makes use of Edmond's Blossom algorithm, to sequentially compute the optimal hybrid (wireless and/or wired) path for each on-demand flow between a given source rack and a destination rack. The main objective is to maximize the throughput of intra-HDCN communications over the wireless and/or wired infrastructure. Mainly, JRCA-HDCN takes into consideration both the i) length of IP queues (waiting delay) in each relay node and ii) level of wireless interferences (retransmission delay). Simulation results, performed for both uniform traffic and real workload collected for Facebook's DC, show that JRCA-HDCN enhances network performance compared to the related strategies.

In our final contribution, we tackle the problem of **Joint Batch Routing and Channel** Assignment (JBRC) in HDCN, to handle the potential batched arrivals of flow requests to the network. The main objective of JBRC problem is to find for each batch of communications, the corresponding hybrid (wireless and/or wired) routing paths. In doing so, an efficient use of wireless and wired resources in the HDCN is ensured. JBRC was formulated as an advanced Multi-Commodity Flow scheme and bears an optimization objective of minimizing the end-to-end delay over all the links of the hybrid routing paths. To solve JBRC, we proposed three main solutions. First, the exact solution, solves the integer linear programming problem JBRC with B&C algorithm, to compute optimal hybrid paths for small instances of JBRC problem. Second, to deal with large instances of JBRC while considering computation time, we proposed a heuristic-based solution JRH-HDCN able to reduce complexity. Third, to ensure a near-to-optimal solution, we put forward an approximate scalable approach SJB-HDCN that considers the dimension challenge and further converges to a feasible solution with a guaranteed precision. Simulation results conducted for uniform traffic pattern as well as Facebook's DC workload traces show that our batch solutions outperforms the online approaches and enhance network performance in terms of total delay and throughput.

7.3 Future research directions

Several future research directions open up. In the following, we will detail the main research work we suggest from a short and a long term views.

First, we have designed, in this thesis, a CLOS-based HDCN architecture, following the MSDC model. Our choice is motivated by the high capabilities of such a model which has shown high performances in real modern DCs. It is straightforward to see that our proposed routing and resource allocation approaches are generic and can be applied to any kind of infrastructure. Unfortunately, this is not the case for several related strategies which are closely dependent on the underlying DCN architecture. Therefore, from a short term view, we will gauge the performance of our strategies with regard to other relevant HDCN architectures, such as the `VL2` architecture [6] propounded by Microsoft, VLCcube [52] and Fat-Tree [62].

In addition, in this thesis, we put forward a centralized controller (CC) that monitors the traffic within the HDCN and decides about the flow routes and channel assignment of on-demand flow requests. Typically, our HDCN can be, actually, considered as an Software Defined Networking (SDN) architecture. The latter is assumed to control both wired and wireless infrastructures making use of a centralized SDN controller. Indeed, it decouples the control plane from the data plane in the DCN, by transforming the switch/routers into simple forwarding devices. These devices have to receive and apply rules sent by the controller using a specific southbound protocol. In our current implementation, the CC has a global view of the network and decides for each flow the proper hybrid (wireless/wired) path. Specifically, when a packet from a flow f arrives to a ToR switch, the next-hop interface is decided by the CC. However, at this stage, we do not make use of SDN controller rules. Instead, the ToR switches forward each packet according to the corresponding interface, without communicating with the CC. Therefore, our next purpose is to extend the OpenFlow protocol [92] so that each hybrid path information (i.e., wireless or wired interfaces) is transformed to specific SDN rules. The latter have to be used by each switch during the forwarding process of the flow. Note that OpenFlow is an open-source southbound protocol commonly used to ensure the interaction between control and forwarding planes.

Moreover, it is worth noting that within the framework of this work, only physical resources have been considered for allocation. In order to provide tenants with virtual networks connecting their compute instances, we aim, in middle term view, to extend the interface between tenants and provider to explicitly consider the network. Actually, regardless of the deployed DCN architecture, connectivity has to be ensured between tenant's VMs allocated on different servers of the network [93]. Therefore, our next objective is to deal with joint Virtual Network Embedding and Routing problem in HDCN. Specifically, we propose to deploy jointly the virtual machine embedding and routing the transmission path simultaneously. Note that the tackled problem is different from the classical virtual network embedding issue in Cloud. Indeed, our future research not only considers the available resources (i.e., CPU, memory) but also has to take into account the congestion level on the ToRs. The proposed algorithm is expected to handle both on-demand and batch request arrivals.

Furthermore, we consider in this thesis only unicast traffic for inter-DCN communications. Actually, recent research directions have started investigating the multicast routing in traditional wired DCN [94] [95]. The main motivation behind the adoption of point-to-multipoint communications in data centers is the massive growth of traffic. Consequently, network layer multicast would help modern product DCNs to save network traffic and to avoid the latency induced by repeated transmissions from the same sender. Therefore, as a future direction, we aim to address the problem of multicast routing in HDCN. The key challenge of such a problematic is to enable the IP multicast fonctionnality, for both control and data planes, in conventional switches and routers, while considering scalability constraint. In fact, tens to hundreds of thousands of servers in the HDCN may participate in the multicast group communication.

7.4 Publications

This section summarizes the publications that have been achieved during this thesis

- **Journals**

  1. Boutheina Dab, Ilhem Fajjari, Nadjib Aitsaadi, "Online-Batch Joint Routing and Channel Allocation for Hybrid Data Center Networks", in IEEE Transactions on Network and Service Management, Special Issue on Advances in Management of Softwarized Networks, August, 2017
  2. Boutheina Dab, Ilhem Fajjari, Nadjib Aitsaadi, "A 2D Beamforming Wireless Resource Allocation Algorithm in Hybrid Data Center Networks", **submitted** in IEEE Journal in Selected Areas on Communications, 2017

- **Conferences**

  1. Boutheina Dab, Ilhem Fajjari and Nadjib Aitsaadi, "A Heuristic Strategy for Joint Batch-Routing and Channel Allocation Approach in Hybrid-DCNs", **submitted** in IEEE GlobeCom 2017, Singapore, December 4-8, 2017
  2. Boutheina Dab, Ilhem Fajjari and Nadjib Aitsaadi "A Joint Batch-Routing and Channel Allocation Approach in Hybrid Data Center Networks", **accepted** in IEEE International Conference on Communications, VTC-Fall 2017, Toronto, Canada, September 24-27, 2017
  3. Boutheina Dab, Ilhem Fajjari and Nadjib Aitsaadi, "A Novel Joint Routing and Channel Allocation Approach in Hybrid Data Center Networks", **accepted** in IEEE International Conference on Sensing, Communication and Networking, SECON, San Diego, USA, Jun, 2017.
  4. Boutheina Dab, Ilhem Fajjari, Nadjib Aitsaadi and Abdehlamid Mellouk, "A Novel Wireless Resource Allocation Algorithm in Hybrid Data Center Networks", **published** in IEEE International Conference on Mobile Ad hoc and Sensor Systems, IEEE MASS 2015, Dallas, USA, October $19-22$, 2015.

- **Technical report**

1. Oussama Soualah, Boutheina Dab, Nadjib Aitsaadi and Abdelhamid Mellouk, "Network functional improvements for large scale IoT deployments", Deliverable 3.4.b, Cloud services - WP3 , Celtic+ TILAS project, September 2015

List of figures

List of tables

References

[1] "5g: A network transformation imperative," tech. rep., Intel.

[2] "Cisco visual networking index: Forecast and methodology, 2015-2020," tech. rep., CISCO, 2016.

[3] M. Bari, R. Boutaba, R. Esteves, L. Granville, M. Podlesny, M. Rabbani, Q. Zhang, and M. Zhani, "Data center network virtualization: A survey," *IEEE COMMUNICATIONS SURVEYS* & *TUTORIALS*, September 2012.

[4] "Mega-datacenter battle update: Internet giants continue to increase capex," tech. rep., August 2016.

[5] D. Abts and B. Felderman, "A guided tour of data-center networking," *ACM Queue Networks magazine*, May 2012.

[6] D. Halperin, S. Kandula, J. Padhye, P. Bahl, and D. Wetherall, "Augmenting data center networks with multi-gigabit wireless links," *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2011.

[7] Y. Zhu, X. Zhou, Z. Zhang, L. Zhou, A. Vahdat, B. Y. Zhao, and H. Zheng, "Cutting the cord: a robust wireless facilities network for data centers," *ACM Conference on Mobile Computing and Networking (MobiCom)*, September 2014.

[8] Y. Cui, H. Wang, and X. Cheng, "Channel allocation in wireless data center networks," *IEEE International Conference on Computer Communications (INFOCOM)*, April 2011.

[9] Y. Cui, H. Wang, X. Cheng, D. Li, and A. Yla-Jaaski, "Dynamic scheduling for wireless data center networks," *IEEE Transactions on Parallel and Distributed Systems*, December 2013.

[10] Y. Cui, H. Wang, X. Cheng, and B. Chen, "Wireless data center networking," *IEEE Wireless Communications*, December 2011.

[11] A. Greenberg, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. Maltz, P. Patel, and S. Sengupta, "Vl2: A scalable and flexible data center network," *ACM SIGCOMM*, August 2009.

[12] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," *ACM SIGCOMM*, August 2008.

[13] "Cisco's massively scalable data center - network fabric for warehouse scale computer," tech. rep., CISCO, December 2012.

[14] S. Hooda, S. Kapadia, and P. Krishnan, *Using TRILL, FabricPath, and VXLAN: Designing Massively Scalable Data Centers with overlays*. CISCO Press, January 2014.

[15] C. E. Leiserson, "Fat-trees: Universal networks for hardware-efficient supercomputing," *IEEE Transactions on Computers*, October 1985.

[16] A. Greenberg, al A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, and P. Lahiri, "Vl2: A scalable and flexible data center network," *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2009.

[17] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu, "Bcube: A high performance, server-centric network architecture for modular data centers," *ACM SIGCOMM*, August 2009.

[18] S. Talebia, F. Alamb, I. Katibb, M. Khamisb, R. Salamab, and G. N. Rouskas, "Spectrum management techniques for elastic optical networks: A survey," *Optical Switching and Networking*, July 2014.

[19] X. Ye, Y. Yin, S. Yoo, P. Mejia, R. Proietti, and V. Akella, "Dos: A scalable optical switch for datacenters," *6th ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, October 2010.

[20] 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," *ACM SIGCOMM*, August 2010.

[21] J. Shin, E. Sirer, H. Weatherspoon, and D. Kirovski, "On the feasibility of completely wireless datacenters," *IEEE/ACM Transactions on Networking (TON)*, October 2013.

[22] "Massively scalable data center (MSDC) design and implementation guide," tech. rep., CISCO Systems, October 2014.

[23] “IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and physical layer (PHY) Specifications Amendment 3: Enhancements for Very High Throughput in the 60GHz Band,” *IEEE Std 802.11ad 2012*, December 2012.

[24] B. Dab, I. Fajjari, N. Aitsaadi, and A. Mellouk, “A novel wireless resource allocation algorithm in hybrid data center networks,” *IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS)*, October 2015.

[25] B. Dab, I. Fajjari, and N. Aitsaadi, “A novel joint routing and channel allocation approach in hybrid data center network,” *IEEE International Conference on Sensing, Communication and Networking (SECON)*, June 2017.

[26] T. Chen, X. Gao, and G. Chen, “The features, hardware, and architectures of data center networks: A survey,” *Journal of Parallel and Distributed Computing*, October 2016.

[27] M. Al-Fares, A. Loukissas, and A. Vahdat, “A scalable, commodity data center network architecture,” *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2008.

[28] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, “Elastictree: saving energy in data center networks,” *The 7th USENIX conference on Networked systems design and implementation*, April 2010.

[29] R. Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat, “Portland: A scalable fault-tolerant layer 2 data center network fabric,” *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2009.

[30] Y. Sun, J. Chen, Q. Liu, and W. Fang, “Diamond: An improved fat-tree architecture for largescale data centers,” *Journal of Communications*, January 2014.

[31] A. Greenberg, J. Hamilton, D. Maltz, and P. Patel, “The cost of a cloud: research problems in data center networks,” *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2008.

[32] D. Abts, M. Marty, P. Wells, P. Klausler, and H. Liu, “Energy proportional datacenter networks,” *ACM SIGARCH Computer Architecture*, June 2010.

[33] M. Csernai, F. Ciucu, R. Braun, and A. Gulyas, “Towards 48-fold cabling complexity reduction in large flattened butterfly networks,” *IEEE INFOCOM*, May 2015.

[34] D. Lin, Y. Liu, M. Hamdi, and J. Muppala, “Flatnet: Towards a flatter data center network,” *IEEE GlobeCom*, December 2012.

[35] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu, “Dcell: a scalable and fault-tolerant network structure for data centers,” *ACM SIGCOMM*, August 2008.

[36] H. Abu-Libdeh, P. Costa, A. Rowstron, G. OShea, and A. Donnelly, “Symbiotic routing in future data centers,” *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2010.

[37] M. Chen, H.Jin, Y.Wen, and V.C.Leung, “Enabling technologies for future data center networking: a primer,” *IEEE Network*, July 2013.

[38] C. Kachris and I. Tomkos, “A survey on optical interconnects for data centers,” *IEEE Communications Surveys and Tutorials*, January 2012.

[39] K. Chen, A. S. A. Singh, K. Ramachandran, L. Xu, Y. Zhang, X. Wen, and Y. Chen, “Osa: An optical switching architecture for data center networks with unprecedented flexibility,” *IEEE/ACM Transactions on Networking*, April 2014.

[40] G. Wang, D. G. Andersen, M. Kaminsky, K. Papagiannaki, T. Eugene, M. Kozuch, and M. Ryan, “c-through: part-time optics in data centers,” *ACM Special Interest Group on Data Communication (SIGCOMM)*, October 2010.

[41] N. Hamedazimi, Z. Qazi, H. Gupta, V. Sekar, S. R. Das, J. P. Longtin, H. Shah, and A. Tanwer, “Firefly: A reconfigurable wireless data center fabric using free-space optics,” *ACM Special Interest Group on Data Communication (SIGCOMM)*, August 2014.

[42] K. Ramachandran, R. Kokku, R. Mahindra, and S. Rangarajan, “60ghz data-center networking: wireless ==> worry less?,” tech. rep., NEC Laboratories America, July 2008.

[43] H. Vardhan, N. Thomas, S. Ryu, B. Banerjee, and R. Prakash, “Wireless data center with millimeter wave network,” *IEEE GlobeCom*, December 2010.

[44] “IEEE Standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements. part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for high rate wireless personal area networks (WPANs) amendment 2: Millimeter-wave-based alternative physical layer extension,” *IEEE Std 802.15.3c-2009 (Amendment to IEEE Std 802.15.3-2003)*, December 2009.

[45] S. Kandula, J. Padhye, and P. Bahl, “Flyways to de-congest data center networks,” *ACM SIGCOMM HotNets*, August 2009.

[46] W. Zhang, X. Zhou, L. Yang, Z. Zhang, B. Y. Zhao, and H. Zheng, “3d beamforming for wireless data centers,” *Hotnets*, November 2011.

[47] X. Zhou, Z. Zhang, Y. Zhu, Y. Li, S. Kumar, A. Vahdat, B. Y. Zhao, and H. Zheng, "Mirror mirror on the ceiling: Flexible wireless links for data centers," *ACM SIGCOMM*, August 2012.

[48] Y. Katayama, K. Takano, Y. Kohda, N. Ohba, and D. Nakano, "Wireless data center networking with steered-beam mmwave links," *IEEE WCNC*, April 2011.

[49] Y. Li, F. Wu, X. Gao, and G. Chen, "Sphericalmesh: A novel and flexible network topology for 60ghz-based wireless data centers," *IEEE/CIC International Conference on Communications (ICCC)*, October 2014.

[50] K. Han, Z. Hu, J. Luo, and L. Xiang, "Rush: Routing and scheduling for hybrid data center networks," *IEEE Conference on Computer Communications (INFOCOM)*, August 2015.

[51] Y. Cui, S. Xiao, X. Wang, Z. Yang, C. Zhu, X. Li, L. Yang, and N. Ge, "Diamond: Nesting the data center network with wireless rings in 3d space," *USENIX NSDI*, March 2011.

[52] L. Luo, D. Guo, J. Wu, S. Rajbhandari, T. Chen, and X. Luo, "VLCcube: A vlc enabled hybrid network structure for data centers," *IEEE Transactions on Parallel and Distributed Systems*, December 2016.

[53] D. Li, C. Guo, H. Wu, K. Tan, Y. Zhang, and S. Lu, "Ficonn: Using backup port for server interconnection in data centers," *IEEE Infocom*, May 2009.

[54] "Analysis of an equal-cost multi-path algorithm," tech. rep., Network Working Group - Request for Comments: 2992, November 2000.

[55] "http://www.sibeam.com/whitepapers/," tech. rep., SiBEAM.

[56] Y. Cui, H. Wang, and X. Cheng, "Wireless link scheduling for data center networks," *5th International Conference on Ubiquitous Information Management and Communication (ICUIMC'11)*, February 2011.

[57] G. Audhya, K. Sinha, S. C. Ghosh, and B. P. Sinha, "A survey on the channel assignment problem in wireless networks," *Wireless Communications and Mobile Computing*, May 2011.

[58] M. P. Mishra and P. C. Saxena, "Issues, challenges and problems in channel allocation in cellular system," *Computer and Communication Technology (ICCCT)*, November 2011.

[59] S. Chakraborty, B. Saha, and S. K. Bandyopadhyay, "Dynamic channel allocation in ieee 802.11 networks," *International Journal of Computer Applications*, September 2016.

[60] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "Distributed channel allocation protocols for wireless sensor networks," *IEEE Transactions On Parallel and Distributed Systems*, September 2014.

[61] K. R. Chowdhurya, N. Nandirajub, P. Chandac, D. P. Agrawald, and Q. Zenge, “Channel allocation and medium access control for wireless sensor networks,” *Elsevier Ad Hoc Networks*, March 2009.

[62] L. Shan, C. Zhao, X. Tian, Y. Chengy, F. Yang, and X. Gan, “Relieving hotspots in data center networks with wireless neighborways,” *IEEE Global Telecommunications Conference (GLOBECOM)*, December 2014.

[63] H. Vardhan, S. Ryu, B. Banerjee, and R. Prakash, “60 ghz wireless links in data center networks,” *Computer Networks: The International Journal of Computer and Telecommunications Networking*, January 2014.

[64] S. Singh, F. Ziliottoy, U. Madhow, E. M. Belding-Royerz, and M. J. W. Rodwell, “Millimeter wave wpan: Cross-layer modeling and multihop architecture,” *IEEE International Conference on Computer Communications, InfoCom*, May 2007.

[65] R. Langar, N. Bouabdallah, R. Boutaba, and G. Pujolle, “Interferer link-aware routing in wireless mesh networks,” *IEEE ICC*, May 2010.

[66] S. Waharte, B. Ishibashi, R. Boutaba, and D. Meddour, “Design and performance evaluation of iar: Interference-aware routing metric for wireless mesh networks,” *Mobile Networks and Applications (MONET)*, December 2009.

[67] S. Waharte, R. Boutaba, Y. Iraqi, and B. Ishibashi, “Routing protocols in wireless mesh networks: challenges and design considerations,” *Multimedia Tools and Applications*, June 2006.

[68] M. Bezahaf, L. Iannone, and M. D. A. S. Fdida, “An experimental evaluation of cross-layer routing in a wireless mesh backbone,” *Elsevier Computer Networks journal*, January 2012.

[69] J. Tang, G. Xue, and W. Zhang, “Interference-aware topology control and qos routing in multi-channel wireless mesh networks,” *The ACM International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc)*, July 2005.

[70] V. Kolar and N. B. Abu-Ghazaleh, “A multi-commodity flow approach for globally aware routing in multi-hop wireless networks,” *IEEE International Conference on Pervasive Computing and Communications (PERCOM)*, March 2006.

[71] L. Xiao, M. Johansson, and S. P. Boyd, “Simultaneous routing and resource allocation via dual decomposition,” *IEEE Transactions on Communications*, July 2004.

[72] A. Islam, M. J. Islam, N. Nurain, and V. Raghunathan, “Channel assignment techniques for multi-radio wireless mesh networks: A survey,” *IEEE Communications Surveys and Tutorials*, December 2015.

[73] A. H. Mohsenian-Rad and V. W. S. Wong, “Joint logical topology design, interface assignment, channel allocation, and routing for multi-channel wireless mesh networks,” *IEEE Transactions on Wireless Communications*, December 2007.

[74] Z. Han, Y. Li, H. Tany, R. Wangz, and Y. Zhang, “Cross-layer protocol design for wireless communication in hybrid data center networks,” *The 12th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, December 2016.

[75] W. Zhang, S. Zhang, Z. Qian, K. Wen, and S. Lu, “Virtual network deployment in hybrid data center networks,” *IEEE Trustcom/BigDataSE/ISPA*, August 2016.

[76] Z. He and S. Mao, “A decomposition principle for link and relay selection in dual-hop 60 ghz networks,” *IEEE INFOCOM*, October 2016.

[77] A. Mehrotra and M. Trick, “A column generation approach for graph coloring,” *INFORMS Journal on Computing*, November 1996.

[78] S. Gualandi and F. Malucelli., “Exact solution of graph coloring problems via constraint programming and column generation,” *INFORMS Journal on Computing*, February 2011.

[79] C. Barnhart, E. L. Jonhson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, “Branch-and-price: Column generation fro solving huge integer programs,” *INFORMS Journal on Computing*, January 1996.

[80] S. Held, W. Cook, and E. Sewell, “Maximum-weight stable sets and safe lower bounds for graph coloring,” *Mathematical Programming Computation*, December 2012.

[81] R. Arjun, Z. Hongyi, B. Jasmeet, P. George, and S. A. C, “Inside the social network’s (datacenter) network,” *ACM Conference on Special Interest Group on Data Communication, SIGCOMM ’15*, August 2015.

[82] A. Abouelaoualim, K. C. Das, L. Faria, Y. Manoussakis, C. Martinhon, and R. Saad, “Paths and trails in edge-colored graphs,” *Theoretical Computer Science*, December 2008.

[83] V. Kolmogorov, “Blossom v: A new implementation of a minimum cost perfect matching algorithm,” *Mathematical Programming Computation*, March 2009.

[84] J. Edmonds, “Maximum matching and a polyhedron with 0-1 vertices,” *Journal of Research at the National Bureau of Standards*, June 1965.

[85] M. L. Fisher, “The lagrangian relaxation method for solving integer programming problems,” *Management Science*, December 2004.

[86] B. Dab, I. Fajjari, and N. Aitsaadi, “A novel joint routing and channel allocation approach in hybrid data center network,” *IEEE International Conference on Sensing, Communication and Networking, SECON*, Jun 2017.

[87] A. E. Ozdaglar and D. P. Bertsekas, “Optimal solution of integer multicommodity flow problems with application in optical networks,” *Symposium on Global Optimization*, June 2003.

[88] E. A. Hansen and S. Zilberstein, “Lao*: A heuristic search algorithm that finds solutions with loops,” *Artificial Intelligence*, March 2001.

[89] J. Castro, “A specialized interior-point algorithm for multicommodity network flows,” *SIAM journal on Optimization*, July 2000.

[90] F. Babonneau, O. du Merle, and J. Vial, “Solving large scale linear multicommodity flow problems with an active set strategy and proximal-accpm,” *Operations Research journal*, February 2006.

[91] “Subgradient methods,” tech. rep., Stanford University, October 2003.

[92] Z. Qin, G. Denker, C. Giannelli, P. Bellavista, and N. Venkatasubramanian, “A software defined networking architecture for the internet-of-things,” *IEEE/IFIP NOMS*, May 2014.

[93] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, “Towards predictable datacenter networks,” *ACM SIGCOMM*, August 2011.

[94] K. Chen, C. Hu, X. Zhangs, K. Zheng, Y. Chen, and A. Vasilakos, “Survey on routing in data centers: Insights and future directions,” *IEEE Network*, July 2011.

[95] O. Komolafe, “Ip multicast in virtualized data centers: Challenges and opportunities,” *IEEE/IFIP IM*, May 2017.