

Use Case-Based Evaluation of Software Defined Networking Controllers



Presented by:
Themba Shozi

Supervisor:
Dr P. Mudali
Dept. of Computer Science
University of Zululand

Co-Supervisor:
Mr. Zaaïd Du Toit
Dept. of Telecoms and Media
Council for Scientific and Industrial Research

Submitted to the Department of Computer Science at the University of Zululand in compliance with the academic requirements for a Master of Science degree in Computer Science

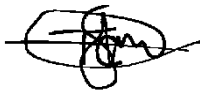
December 13, 2019

In memory of my father (Gabhisa, Mtumaseli, Mbasha, Khuluse).

”Singabakwa Shozi ngenkani izwe silihambile”.

Declaration

1. I understand that plagiarism is a violation. Plagiarism is about using the work of someone and claiming it as your own.
2. The citation and references used in this work are in accordance with the IEEE convention. Every contribution to this document from other people's work(s) has been credited, cited and referenced.
3. The work presented in this dissertation is originally mine.
4. I have not permitted any individual to duplicate my work, or part of it, with the purpose of presenting or transferring it as their own work.



—
Themba Shozi
December 13, 2019

Acknowledgments

Profound gratitude to my mentor, former manager and supervisor Sabelo Dlamini. Thank you so much for your motivation, patience, enthusiasm and willingness to share your extensive knowledge of networking and life in general. Your passion and dedication to develop young people are humbling (the “formula” is working). I am forever beholden to you.

To my supervisor Dr Pragasen Mudali. Thank you so much for your willingness to assist me whenever I needed your assistance and for your constant faith in me.

To my sister Mpume Manqanqa and her husband Bongani Manqanqa. Thank you so much for believing in my dreams and for putting aside your own dreams just to see your young brother succeed. Most importantly, I sincerely appreciate your financial support and words of motivation. You truly are a great sister and I will forever be thankful. Bongani I will always remember those prayers we had every time you dropped me off at school. Thanks, Brother.

To all my friends and family, (the list will cause conflict) brothers and sisters, I appreciate all the support and motivation.

To the Queen mother, I would like to convey my heartfelt love to my dear mother (Khathide, Sobahle, Njenje) for her continuous words of encouragement and unfailing support. I will forever be grateful for what you have done for me and our family.

The authors appreciate the support of the Council for Scientific and Industrial Research (CSIR) with financial assistance and the use of their facilities to conduct this research, the South African National Research and Education Network (SANREN) for providing network data and the University of Zululand for academic and technical support for this study.

Research Outputs

The study produced the following publications:

1. Themba Shozi, Sabelo Dlamini, Pragasen Mudali, and Matthew O. Adigun. “An SDN Solution for Performance Improvement in Dedicated Wide-Area Networks.” At 2019 Conference on Information Communications Technology and Society (ICTAS), pp. 1-6. IEEE, 2019.
2. Themba Shozi, Sabelo Dlamini, Pragasen Mudali, and Hlabishi Kobo, “SDN-enabled Switch Placement in Multi-Domain Hybrid SD-WAN” Accepted for 2019 IEEE AFRICON, Ghana, Accra, pp. 1-6. IEEE, 2019.
3. Themba Shozi, Pragasen Mudali, Matthew O. Adigun and Hlabishi Kobo. “A Data Plane Multi-Path Load Balancing Mechanism for Hybrid SDN in Different Topologies”, At 2019 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), South Africa, Drakensberg Sun Resort, Winterton, KwaZulu Natal, pp. 1-6. IEEE, 2019.
4. Themba Shozi, Sabelo Dlamini, Pragasen Mudali, and Matthew O. Adigun. “An Economic Analysis of Hybrid SDN as an Enabler in African Software-Defined NRENs”, At Southern Africa Telecommunication Networks and Applications Conference (SATNAC), South Africa, Zimbali Resort in Ballito, KwaZulu-Natal, pp. 1-6, 2019

Contents

Chapter 1.....	1
-----------------------	----------

Introduction	1
---------------------------	----------

1.1 Preamble.....	1
-------------------	---

1.2 Problem Statement	4
-----------------------------	---

1.3 Motivation.....	5
---------------------	---

1.4 Research Questions.....	7
-----------------------------	---

1.5 Research Goal and Objectives.....	7
---------------------------------------	---

1.5.1 Research Goal.....	7
--------------------------	---

1.5.2 Research Objectives.....	7
--------------------------------	---

1.6 Research Approach	8
-----------------------------	---

1.7 Scope and Limitations	9
---------------------------------	---

1.8 Dissertation Organization.....	9
------------------------------------	---

Chapter 2	12
------------------------	-----------

Literature Review	12
--------------------------------	-----------

2.1 Preamble.....	12
-------------------	----

2.2 Hybrid Software Defined Networks	12
--	----

2.3 Network Overlay	16
---------------------------	----

2.4 Switch Placement.....	17
---------------------------	----

2.5	Load Balancing.....	19
2.5.1	Economic Analysis.....	21
2.6	Chapter Summary	26
Chapter 3		27
Methodology		27
3.1	Preamble.....	27
3.2	Applied Methodologies	27
3.2.1	Quantitative Research	29
3.2.2	Qualitative Research.....	29
3.3	Implementation	30
3.3.1	Simulation Tools.....	31
3.3.2	Test Environment and Tool Setup.....	31
3.4	Analysis.....	34
3.5	Chapter Summary	34
Chapter 4		34
Hybrid SD-WAN Overlay.....		34
4.1	Preamble.....	34
4.1.1	Assumptions.....	34
4.2	Proposed Solution Overview	35

4.3	Flexibility Evaluation	38
4.3.1	Controller throughput	38
4.3.2	Controller scalability	39
4.3.3	Controller latency	40
4.4	Reliability Evaluation	41
4.5	Chapter Summary	46
Chapter 5	48
Hybrid SD-WAN Switch Placement	48
5.1	Preamble	48
5.1.1	Assumptions.....	48
5.2	Algorithm(s)	48
5.2.1	Agglomerative Hierarchy Clustering	48
5.2.2	Optimized K-means.....	49
5.2.3	Dunn's Index	50
5.3	Algorithm Evaluation	50
5.4	Performance Testing	52
5.5	Simulation Experiment.....	55
5.6	Chapter Summary	57
Chapter 6	59

Multi-Path Load Balancing.....	59
6.1 Preamble	59
6.2 Proposed Solution Overview	60
6.2.1 Scenario 1: Fat Tree Hybrid SDN	61
6.2.2 Scenario 2: Ring Hybrid SDN	63
6.2.3 Scenario 3: Mesh Hybrid SDN.....	66
6.2.4 Scenario 4: Torus Hybrid SDN	69
6.3 Chapter Summary.....	72
 Chapter 7	 74
 Hybrid SDN as an Enabler for African SD-WANs	 74
7.1 Preamble	74
7.2 Proposed Solution	75
7.3 Quantitative Evaluation.....	77
7.4 Recommendations.....	79
7.5 Chapter Summary.....	80
 Chapter 8	 81
 Conclusion and Future Work	 81

List of Figures

1.1	<u>High-level SDN reference architecture</u>	2
1.2	<u>Hybrid SDN Architecture</u>	6
2.1	<u>Volatility Index for Emerging Markets (Santacreu, 2015)</u>	21
3.1	<u>Methodology Framework</u>	26
3.2	<u>Simulation tools</u>	29
3.3	<u>Setup of the benchmarking experiment</u>	30
4.1	<u>SANReN Metros (SANReN CA, 2018)</u>	35
4.2	<u>SDN Overlay WAN Architecture</u>	36
4.3	<u>Throughput vs Switch Graph</u>	37
4.4	<u>Throughput vs MACs Graph</u>	38
4.5	<u>Average latency under switches</u>	39
4.6	<u>Average latency under MACs</u>	40
4.7	<u>Ping test results (Non-edge)</u>	41
4.8	<u>Ping test results (Edge)</u>	42
4.9	<u>Ping test results (Non-edge)</u>	43
4.10	<u>Ping test results (Edge)</u>	43
4.11	<u>Bandwidth test for edge and non-edge</u>	44

5.1	<u>Validity Index Graph</u>	51
5.2	<u>Processing Time Graph</u>	52
5.3	<u>K-means Cluster</u>	53
5.4	<u>AHC Cluster</u>	54
5.5	<u>Controller performance test</u>	55
5.6	<u>Round trip time performance results</u>	56
5.7	<u>Throughput performance results</u>	57
6.1	<u>Fat tree topology</u>	61
6.2	<u>Fat tree Round Trip Time</u>	62
6.3	<u>Fat tree Round Iperf</u>	63
6.4	<u>Ring topology</u>	64
6.5	<u>Ring topology Round Trip Time</u>	65
6.6	<u>Ring topology iperf</u>	66
6.7	<u>Mesh topology</u>	67
6.8	<u>Mesh topology Round Trip Time</u>	68
6.9	<u>Mesh topology Iperf</u>	69
6.10	<u>Torus topology</u>	70
6.11	<u>Torus topology Round Trip Time</u>	71
6.12	<u>Torus topology Iperf</u>	72
7.1	<u>Project Plan Gantt Diagram</u>	77
7.2	<u>Cost Estimation Graph</u>	7

List of Tables

<u>2.1</u>	<u>Feature-based comparison of open-source SDN controllers (Mamushiane, Lysko and Dlamini, 2018b).</u>	12
<u>2.2</u>	<u>High-level comparison of southbound protocols (Mamushiane, Lysko and Dlamini, 2018b).</u>	13
<u>2.3</u>	<u>Comparison between traditional networks and SDN (Singh and Srivastava, 2018)</u>	14
<u>2.4</u>	<u>Literature Gaps Summary</u>	24
<u>7.1</u>	<u>Hybrid CAPEX and OPEX recovery</u>	79
<u>A1</u>	<u>Total average latency for one controller</u>	96
<u>A2</u>	<u>Total average latency for two controllers</u>	96
<u>A3</u>	<u>Effect of switch placement on controller performance</u>	97
<u>A4</u>	<u>Benchmarking results under a varying number of switches</u>	98
<u>A5</u>	<u>Benchmarking results under a varying number of MACs</u>	99

List of Acronyms

ICT	Information and Communication Technologies
CapEx	Capital Expenditure
SDN	Software Defined Networking
NFV	Network Function Virtualization
EC	Edge Computing
ETSI	European Telecommunications Standards Institute
API	Application Programming Interface
IoT	Internet of Things
WAN	Wide Area Network
SANReN	South African National Research Network
2G	Second Generation of Mobile Communications
5G	Fifth Generation of Mobile Communications
WAN	IP Multi-media Subsystem
NREN	National Research and Education Network
TCP	Transmission Control Protocol
BGP-LS	Border Gateway Protocol Link State
LISP	Locator ID Separation Protocol
I2RS	Interface to Routing System
OVSDB	Open vSwitch Database Management
NETCONF	Network Configuration
OF-CONFIG	OpenFlow Configuration
PCEP	Path Computational Element Protocol
TLS	Transport Layer Security
ONOS	Open Network Operating System
MDSE	Model-Driven Software Engineering Principle
MD-SAL	Model-Driven Service Abstraction Layer
LSO	Lifecycle Service Orchestration
MEF	Metro Ethernet Forum
S3P	Security, Scalability, Stability and performance
SAL	Service Abstraction Layer
ONAP	Open Networking Automation Platform
GML	Geography Markup Language
PoP	Point of Presence

Abstract

The paradigm of Software Defined Networking (SDN) model has arisen as a viable candidate to tackle traffic management and costs based problems in developing markets. At the root of SDN is the control system that responsible for overseeing resource orchestration premised around its global understanding of the existing resource utilization of the network. A significant problem that needs to be resolved in the early implementation of an SDN-based network configuration is where each controller and switches are located to ensure the optimal operational performance. It is critical to note that the placement and size of the data plane devices have a direct impact on the overall performance of the SDN controller. This problem represents conflicting priorities, such as delay, dynamic routing and efficiency. In order to address the controller performance problem and formulate a strategy for the deployment of SDN in emerging markets, the study proposes solutions that provide flexibility, ease of management, traffic engineering and network distribution with the aim of reducing delay and of enabling efficient network utilization.

There are important questions that should be answered during transitioning to an SDN solution, and those questions are: Given a legacy WAN, how many SDN controllers and switches are required? How should they be placed? Where are the best locations to place them? And how efficient are they in terms of traffic engineering so that the overall network performance is optimized? Henceforth, we will refer to these questions as the switch-to-controller performance problem. The average latency was reduced to 15% compared to that of the Optimized K-means. The proposed method was also successful in the throughput (with an improvement of 13%).

The studied metrics on RTT and throughput showed improvement in the reliability test equal to averages of 1.182 and 2.166 ms with two and three-controller deployment respectively and a throughput average of 20 GB/s. The results have displayed a positive impact in terms of improved controller performance on the selected use-cases. The tree topology yielded an improved average RTT of 0.4 ms and 0.2 to 0.3 GB/s in transfer speed and 0.2 Gbps in bandwidth utilization in Section 6.2.1. The ring topology resulted in an average of 0.6 ms RTT, the transfer speed of 1.2 GB/s and bandwidth of 0.2 Gbps in Section 6.2.2. The mesh topology resulted in an average RTT test of 0.5 ms, with a transfer speed of 0.9 GB/s and a 0.1 Gbps of bandwidth utilization in Section 6.2.3. The Torus topology in RTT yielded an average of 0.6 ms in Section 6.2.4. The transfer speed improved by 0.1 GB/s and bandwidth utilization was not very efficient at 0.2 GB/s. This study can be used by network engineers as a foundation to start integrating SDN or plan a new SDN deployment, by assisting engineers to make quick decisions about optimal switch placement and SDN deployment.

Chapter 1

Introduction

1.1 Preamble

Software-Defined Wide Area Networking (SD-WAN) is a paradigm that permits the supply of numerous networking services on a larger scale of connected networking gadgets. SDN implements ease of management, less expensive telecom setup with robust traffic engineering and a global view of the network and application methods. The controllers carry out tasks in different ways for every network setup by relying on the size and the topology type (Alrashedy et al., 2017). The SDN architectural style offers better network governance and a setup that can minimize OPEX, and improved performance and productivity of telecommunications (Wang et al., 2017). SDN's objective is to achieve agility and flexibility by separating the network control functionality from the forwarding device and locating it on a logically consolidated controller. Centralized SDN control and management can help to allocate network capacity more efficiently, adapt dynamically to changing demand, and deal more rapidly with failure (Bispo et al., 2017), alternatively reducing the over-provisioning and network complexity in WANs.

The diagram shown in Figure 1.1 represents the SDN architecture, which drives the design implementation of SDN controllers.

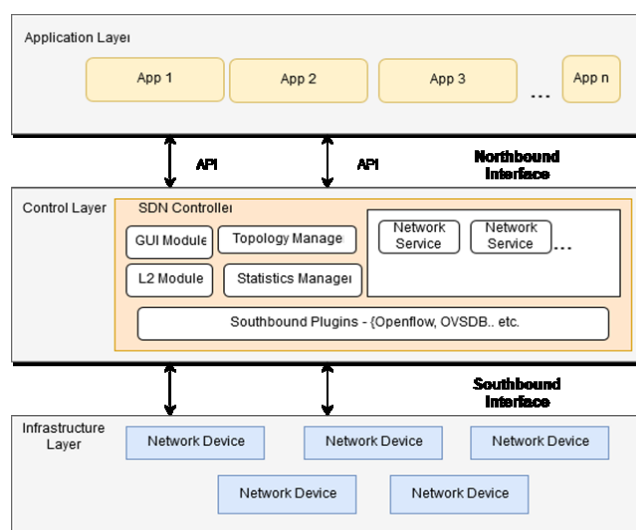


Figure 1.1: High-level SDN reference architecture.

Figure 1.1 shows the architecture of a software-defined network, from the underlying network devices to the applications that run on top of the SDN controller.

The controller handles all communications across applications and devices to modify and manage the flows on the network explicitly. The controller acts as a broker between the applications and the underlying forwarding hardware. The controller enables network administrators to manage traffic in the network in a more granular and dynamic fashion. SDN controllers come in all shapes and sizes and they are tailored for different functions and use cases, the essence of SDN is always kept in check, which is to decouple the network control from the data plane in the underlying hardware architecture. Table 2.4 extracted from (Mamushiane, Lysko and Dlamini, 2018b) displays different open-source SDN controllers that this study will use, as well as their advantages and drawbacks.

Throughout the years WANs have evolved and a number of different routing techniques have been adopted, but CISCO's Multiprotocol Label Switching (MPLS) seems to stand out from the rest (Aerohive, 2017). However, MPLS has its own cost and bandwidth drawbacks. The connections are fairly expensive, offering a low bandwidth when you compare it to public internet services such as IPsec Virtual Private Network (VPN), that share secure information for companies on the public internet (Mitchell, n.d.). MPLS also poses problems for service providers as most companies are shifting to cloud services that are public. These problems occur when service providers attempt to connect company branch networks to third-party data centres through the public cloud (S'eremet and C'au'sevi'c, n.d.). Despite the technical issues presented by MPLS, it is trusted by clients and service providers, because MPLS has been considered to be robust and secure for over 16 years, but the replacement of the old with new equipment becomes a huge financial constraint on the side of the service provider and the customer (Manel et al., 2017). Understanding and operating a new model or architecture can be costly in terms of manpower and scarcity of people with the required skills. The above issues compel both companies and service providers to seek alternatives or better solutions.

WANs are characteristically expensive, and they come with complex management capabilities as they are mostly proprietary (Patrushev and Drozdova, 2017). The costly high-performance routers directly impact the capital expenditure (CAPEX) and the complex management directly affects the operational expenditure (OPEX) and increased probability of failure of the organizations that provide and use the services provided by these legacy networks (Michel and Keller, 2017).

In addition, the over-provisioning of WANs, due to an increased number of users' day-to-day demands, makes them prone to failure. But SDN introduces new networking capabilities with the potential of resolving some of the legacy networking problems. The main intent of SDN is agility and flexibility by separating the network control capabilities from the hardware device and placing these on a logically centralized controller. The controller's responsibility is to program forwarding to the switches through standard protocols, for example, OpenFlow ([Alsaeedi et al., 2019](#)). The SDN drive is to implement a network operating system that performs networking tasks without additional software elements and permits application development that controls the network hardware on a network operating system ([ONF, 2012](#)).

SD-WAN will change the WAN marketplace by leveraging the SDN principles with the escalated need for more bandwidth that comes with next-generation technological advancements such as artificial intelligence (AI) ([Guo et al., 2018](#)), 5G ([Zaidi et al., 2018](#)) and Internet of Things (IoT) which are expected in the 2020s ([Tello-Oquendo et al., 2018](#)). As business will need more flexible, cloud-ready and easily managed network services that are scalable and cost-effective, SD-WAN is tailored not only to connect sub-networks (Local Area Networks (LAN)), such as the legacy MPLS WAN but also to deliver faster connectivity with minimal configuration complexities through the separation of the control plane from the data plane, thus allowing centralized configuration and ease of management ([Lee et al., 2018](#)). This work focuses on the SDN data plane layer as an enabling solution to joining the technological divide in emerging markets to ensure a smooth transition to the future of software-defined technologies by evaluating controller performance and using the hybrid SDN approach in WAN (particularly NRENs) for emerging markets as a use case.

1.2 Problem Statement

The problem of choosing a controller may be as granular as figuring out why a single node is not working correctly or as big as implementing an organization's software solution to streamline operations ([Belkadi and Laaziz, 2017](#)). It all depends on the use case requirements and the business functions of each group or individual. WANs are usually costly and problematic to manage ([Wang, 2018](#)). They have dedicated and costly high-performance routers with high capital expenditure (CAPEX), and complex management that require extensive management that increases operational expenditure (OPEX).

The IETF draft delivers non-specific procedures for benchmarking SDN controller performance by considering an SDN controller as a black box, irrespective of the architecture, application and use case ([Vengainathan et al., 2018](#)). For that reason, we need to find out which SDN controller from the prevalent open-source selections is most suitable [Mamushiane, Lysko and Dlamini, 2018b](#). How many SDN controllers are required in a WAN and where should the SDN switches be placed optimally to meet end-user requirements and limitations while upholding satisfactory performance and quality of service? Additionally, how much it will cost to deploy them together with the manpower involved with the construction and operation of such networks ([Caria et al., 2013](#); [Gu et al., 2017](#))? This is a multi-dimensional optimization problem and comprises contending objectives. Therefore, it is essential to solve this problem during the early phases of SDN migration planning ([Jia et al., 2016](#); [Kar et al., 2016](#)).

There is a gap in studies exploring detailed solutions to quantifying and positioning SDN controllers relative to the switches in the WAN and the analysis of the SD-WAN using the Hybrid SDN approach on a real network scenario with load-balancing techniques suited to hybrid SD-WAN. However, a few authors have reviewed the highest level of development of hybrid SDN networks. [Fernando et al. \(2015\)](#) and [Thyagaturu et al. \(2016\)](#) described various typical hybrid SDN techniques, but without considering the study of hybrid SDN networks. Therefore it is rather a crucial case of SDN, as a separate research area addressing open issues and study limitations in ([Vissicchio et al., 2014](#)). Nevertheless, no concrete strategies are provided in emerging markets for SDN networks. [Rathee et al. \(2019\)](#) reviewed and investigated several deployment techniques in hybrid SDN networks. However, these studies do not appraise the optimization plan and techniques in hybrid SDN networks specifically for emerging markets. In view of the abundance of controllers, there is also a need for assessment in different realistic network scenarios to provide an objective conclusion concerning the controller that would be suitable for the deployment.

1.3 Motivation

The rapid transition in the WAN market triggers the need for a strategic approach to deploy the SDN technology steadily and optimally to ensure its survival and positive influence in the market. WAN's single MPLS connections surrendered to hybrid MPLS and broadband Internet links connecting to cloud as they have shifted their resources. This transformation entails problems for network administrators who are expected to counter the rapidly increasing demands of these networks ([Aerohive, 2017](#)). The migration to SDN to leverage the Network Function Virtualization (NFV) ([Virtualisation, 2012](#)) capabilities which bypass the legacy hardware by decoupling network functions to facilitate management and orchestration (MANO) ([Mijumbi et al., 2016](#)) and service function chaining will unlock network slicing capabilities and secure network services ([Trajkovska et al., 2017](#)).

To date, there has been a proliferation of SDN controllers intended to address diverse use case scenarios warranting the need to evaluate controller performance. Vodafone has reported that they observed development in performance across the network infrastructure after they had deployed an SD-WAN solution with the Industrial Development Corporation (IDC) suggesting a compound annual growth rate (CAGR) of 53.9% by the year 2020 ([Marcus and Molnar, 2017](#)). As far as we could ascertain, little or no study has undertaken a detailed Hybrid SDN migration approach to network southbound layer benchmarking and architectural analysis of the economic implications the technology. Therefore, there is a need for a study that highlights the technical and financial implications of migration to SDN for emerging markets.

Owing to the lack of studies featuring the steady strategy of Hybrid SDN in [Figure 1.2](#) as an enabler of the next-generation network, through the optimal switch and controller placement for emerging markets as a use case ([Huang et al., 2018](#)). This work is tailored to facilitate the deployment of a hybrid transitioning strategy to SDN in emerging markets and intends to study several approaches to the development of an optimum hybrid SDN migration strategy to accelerate SDN deployment in emerging markets.

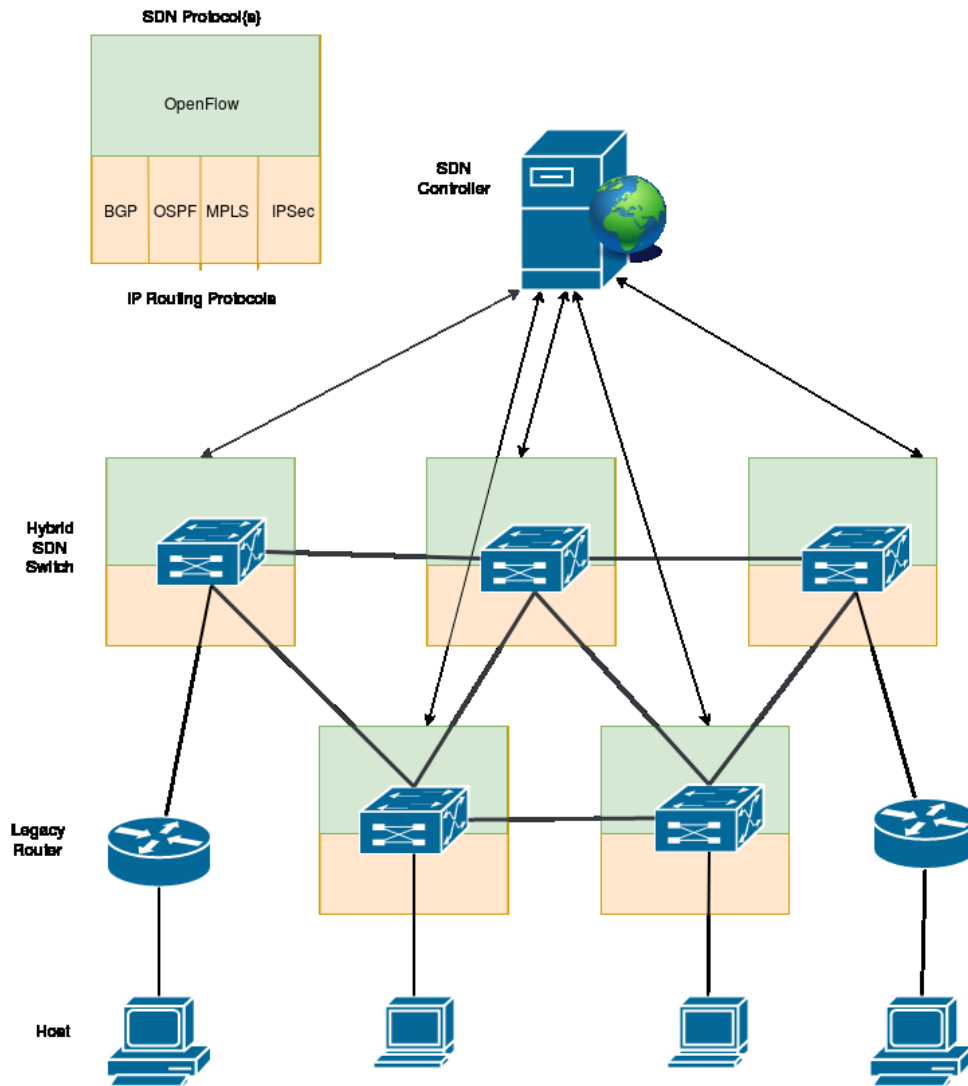


Figure 1.2: Hybrid SDN Architecture

Unlike the existing studies which have only constructed mathematical models, this study takes advantage of simulation and mathematical modelling to address the switch placement problem and develop a deployment technique for hybrid SDN. Like the controller placement problem, the switch placement problem is also considered to be a planning problem and it is not time-sensitive. Therefore, numerous machine learning techniques are being explored to decide on the number of controllers and switches required for a particular hybrid SD-WAN size and different architectural designs. Furthermore, a comparative controller performance evaluation for objective selections and economic quantitative assessment is conducted, together with the financial assessment related to the migration to SDN by using the hybrid SDN technique in emerging markets.

1.4 Research Questions

The research questions this work aims to address are:

- How does the number and placement of SDN switches impact the performance of the controller in SDN-enabled WANs?
- What mechanisms can be used to manage workload distribution across the SD-WAN?
- What are the financial implications associated with the migration to SDN-enabled WANs?

1.5 Research Goal and Objectives

1.5.1 Research Goal

The goal evaluate the performance of SDN controllers in different application scenarios.

1.5.2 Research Objectives

The objectives of this work are as follows:

1. To investigate the current status of controller performance and the evaluations in different application scenarios through literature
2. To investigate the influence of switch placement on the network performance by means of mathematical modelling and simulation;
 - To determine the suitable number of switches to use, given a network's size;
 - To determine the optimal location for switches in an SDN-enabled WAN;
3. To investigate a mechanism to manage workload distribution;

4. To investigate economic impacts of migrating to SDN-enabled WAN

1.6 Research Approach

In order to make an informed decision about a resourceful open-source SDN controller, a pre-emulation experiment for performance comparison of SDN controllers is required. To improve switch placement in WANs, partitioning machine learning algorithms are studied. The intention is to take advantage of network mathematical modelling to address the switch placement problem by investigating the optimal number and location(s) for switch placement. To ensure the validity of the mathematical model results and assess the applicability of real WAN deployment, the switch placement problem is investigated by using an emulation platform to emulate a realistic network and a real open-source SDN controller. This assessment comprises contending objectives such as throughput, latency and scalability.

Secondly, the study investigates mechanisms that can be used to manage data plane workload distribution during the migration to SDN in the event of congestion and link failure (during and after network failure). Several load-balancing techniques are studied and an alternative multi-path technique for a hybrid SDN-enabled environment is developed and tested to determine how it characterizes different paths for a packet flow and how it shares traffic load evenly among distinct flows. To ensure the validity of the mathematical model, using an emulation platform and a real open source SDN controller, resourcefulness of the load balancer is measured by using four distinct topologies, specifically, Fat Tree, Ring, Mesh and Torus topology.

Lastly, the study develops practical guidelines for a multi-objective SDN network migration plan. The aim is to provide a financial migration plan and investigate the economic impact of migrating to SDN-enabled WAN using a financial model which considers emerging markets by using capital budgeting to assess the quantifiable significance associated with the migration. The study explores a more refined model that considers the time value of money, and the impact of SDN on the CAPEX and OPEX is quantified for the government business case, as it is considered to be the primary source of funding because it is responsible for capital expenses for primary setups and future upgrades.

1.7 Scope and Limitations

One key limitation of this work is a lack of the necessary equipment for testbed evaluation of our proposed approaches. Ideally, we would like to deploy physical Ubuntu machines in different geographic locations to mimic a real WAN topology, utilizing a tool such as Cbench (*cbench*, n.d.) to perform complex analysis of the switch placement problem. However, owing to budgetary constraints, this work relies on simulation and emulation to address the switch placement problem. Prototypes built on top of an emulated environment can easily be moved to a real network environment with minimal changes to the codebase. SDN has its own limitations, with the OpenFlow protocol (Benabbou et al., 2019) that is not purely matured to support commercial SDN technology and also switches and controllers are not entirely dependable and established (Rojas et al., 2018). The significant performance metrics used to study network performance include network latency, network throughput, network scalability, network reliability, and traffic load control.

Another challenge involves limited options in terms of dataset regarding emerging-market topologies (particularly African). Unfortunately, the only datasets recorded for African backbones are SANREN (a South African NREN) and ZAMREN (a Zambian NREN), but the ZAMREN offers minimal details of the network. Nevertheless, the proposed approaches are generic and can be used to optimize other topologies of different sizes and configurations.

1.8 Dissertation Organization

The remainder of this dissertation comprises the following: Chapter 1 offers the introduction to the study, while Chapter 2 presents a literature review of traditional networks, SDN technology as well as current research work on the switch placement problem and WAN. Chapter 3 presents methodologies that are undertaken in this study to gather research information to deliver a solution to the formulated problem. Chapter 4 presents a WAN environment SDN overlay solution in emerging markets. The results from and discussions of this evaluation are included in the same chapter. Chapter 5 feeds on the results from Chapter 4 by proposing an AHC method integrated with a cluster validity measure to determine the clusters on the network to address the switch placement problem and eliminate the inconsistency related to other placement techniques. Chapter 6 considers the results from Chapter 4 and 5 by conducting a load-balancing test

of four separate hybrid topologies using a network data plane multi-path load-balancing system to reduce control plane overhead, and presents results generated by implementing these techniques. In order to address the question of the capital and operational expenditures involved in the migration, Chapter 7 presents a techno-economics analysis with the focus on the data plane as a primary asset with the aim to migrate to Hybrid SDN. Lastly, Chapter 8 concludes this study and highlights lessons learnt and future directions for further research.

Chapter 2

Literature Review

2.1 Preamble

This chapter focuses on the detailed literature and analysis of related studies in order to gather research information for this study. The information will be used to solve the problem presented in Chapter 1 [section 1.2](#). The literature is divided into two subsections, first, the background of the study at hand and the related literature that provided the foundation on which this work has been built. The literature in this section includes quantitative and qualitative theoretical and practical results of the previous work and then presents the shortcomings of several studies that will help solidify the need for this study and show how each objective will tackle these shortcomings. This section will help us to gather and analyze both primary and secondary data.

2.2 Hybrid Software Defined Networks

Several studies on the analysis of a solution to the WAN problems have revealed that a new paradigm known as Hybrid SDN can provide a solution ([Index, 2017](#); [Marcus and Molnar, 2017](#); [Rosendo et al., 2017](#)). Hybrid SDN is a type of network architecture where both traditional and SDN devices coexist and communicate in order to achieve better configuration, central and distributed control and management of network behaviour for optimal network performance and user experience. This is possible through legacy devices consisting of distributed algorithms such as interior gateway protocols ([Kouicem et al., 2017](#)) which are capable of controlling the overall traffic routing whilst the controller channels traffic from a global point of view in SDN ([Hakiri et al., 2017](#)). [Table 2.4](#) presents different features of several open-source SDN controllers. The most important outcome of going hybrid is the ability of organizations to migrate gradually to the paradigm of SDN, without complete, expensive and extensive replacement of existing and functioning network infrastructure.

Table 2.1: Feature-based comparison of open source SDN controllers (Mamushiane, Lysko and Dlamini, 2018b)

Feature	ONOS	OpenDayLight	Floodlight	Ryu
Southbound Interfaces	OF 1.0, 1.2, 1.3, 1.4, 1.5, NETCONF	OF 1.0, 1.2, 1.3, 1.4, 1.5, NETCONF/YANG, OVSDB, LISP, BGP-LS, SNMP PCEP	OF 1.0, 1.2, 1.3, 1.4, 1.5	OF 1.0, 1.2, 1.3, 1.4, NETCONF, OVSDB, OF-CONFIG
REST-API	Yes	Yes	Yes	Yes
GUI	Web-based	Web-based	Web-based	Yes (Initial phase)
Modularity	High	High	Medium	Medium
Orchestrator Support	No	Yes	Yes	Yes
OS Support	Linux, MAC, Windows	Linux, MAC, Windows	Linux, MAC, Windows	Most supported on Linux
Partner	ON.LAB, Cisco, Huawei, Ericsson, Nec, Ciena Fujitsu, Sk Telecom	Linux Foundation with members covering over 40 companies such as Cisco, IBM, ect.	Big Switch Networks	Nippon Telegraph and Telephone Corporation
Documentation	Good	Very good	Medium	Good
Programming Language	Java	Java	Java	Python
Multi-threading Support	Yes	Yes	Yes	Yes
TLS Support	Yes	Yes	Yes	Yes
Virtualization	Mininet and OVS	Mininet and OVS	Mininet and OVS	Mininet and OVS
Application Domain	Data center and SD-WAN	Data center and SD-WAN	Campus	Campus
Distributed/Centralized	Distributed	Distributed	Centralized	Centralized

To accomplish optimized network core traffic engineering with more inventive use of expensive and complex computing resources, the global network view and control of forwarding decisions that SDN offers can indeed be beneficial in WANs. Both southbound protocols (PCEP and BGP-LS) also offer the ability to explore SDN rollout approaches such as Hybrid SDN. Using these organizations can upgrade their existing traditional networks to programmable SDN. Table 2.2 below provides a highlight of the SDN southbound protocols.

Table 2.2: High-level comparison of southbound protocols (Mamushiane, Lysko and Dlamini, 2018b)

Protocol	Standardization Body	Purpose	Application Domain	Interface Name
OF-CONFIG	ONF	Management	SDN	Southbound
NETCONF	IETF	Management	SDN Hybrid SDN	Northbound, Southbound, East/Westbound
BGP-LS	IETF	Control	Hybrid SDN	Southbound East/Westbound
LISP	IETF	Control	Hybrid SDN	Southbound
OVSDB	ETSI	Management	SDN	Southbound
I2RS	IETF	Control	Hybrid SDN	Southbound
PCEP	IETF	Control	Hybrid SDN	Southbound
OpenFlow	ONF	Control	SDN	Southbound

Table 2.3 summarizes the traditional and software-defined networks and shows the advantages of SDN over traditional networks on various features by using the feature analysis conducted in this literature section. The SDN software requires a qualified network operator to define new policies, however, there are currently no manufacturing-grade approaches to gradual implementation (Sinha et al., 2017). In a scenario where SDN demands a massive transformation to the network model, mainly on architectural updates on the part of the operators, network engineers are required to learn and develop skills for designing, updating, debugging and operating the new network architecture, therefore organizations will need to employ skilful SDN network engineers (Horvath et al., 2015). The shift in market-related dynamics could therefore only be a concern for service providers to provide end-users with a seamless operation (Vissicchio et al., 2017).

Table 2.3: Comparison between traditional networks and SDN (Singh and Srivastava, 2018)

Criteria	Traditional Networks	SDN
Network management and configuration	Require the use of vendor-specific commands which make it difficult to program changes and require specialized training	Enables network programmability by exposing vendor-agnostic interfaces
Global network state awareness	Difficult owing to vertical integration between control and data plane	Simplified through decoupled logically centralized controller
Maintenance cost	Higher	Less
Time required for upgrades/error handling	Sometimes takes months	Can take as little as a few minutes due to centralized control logic
Control plane load balancing	Not important	Important
Control plane utilization	Not relevant	Important
Control plane availability	Not important	Critical
Resource utilization	Less	High
Flow table and state of information integrity	Critical	Important
Control plane integrity, authenticity and consistency	Not relevant	Important

SD-WAN has a number of technical questions to be responded to such as how a centralized controller can be scalable, resilient and robust without being the distinct point of failure (Sinha et al., 2017). The steady migration to new SDN can be helpful when it comes to avoiding potential networking problems and blackouts while delivering the benefits of SDN control and management capabilities, such as centralized control, increased flexibility, enhanced security, and quality consumer experience. The management takes place only through the distributed-control plane of conventional devices (Xu et al., 2018), an SDN switch understands legacy procedures such as LLDP and STP (Alrashedy et al., 2017). This necessitates the management of traffic in the Hybrid SDN data plane layer by exploring techniques such as load balancing and switch placement. The Open Network Foundation (ONF) document highlights hybrid SDN in the section titled Traditional Networking Coexistence and Migration (ONF, 2015). The document provides details that show that hybrid SDN networks are a feasible and tolerable solution for organizations to migrate in the direction of SDN. The document furthermore discusses setups and concerns that need to be dealt with in order to safeguard fruitful migration to coexist with legacy networking technologies. The IETF informational request for comments (RFC) stated that the SDN paradigm is expected to be deployed gradually throughout numerous network and service sections (Boucadair and Jacquenet, 2014). The document also recognises that the SDN architecture is also likely to concur with the traditional networks as hybrid SDN networks as illustrated in Figure 1.2.

Large National Research and Education Networks (NREN), such as G'EANT, are still based on legacy infrastructures that operate on top of a bolted and costly control plane and data plane (Ventre et al., 2017). But the SDN-enabled devices are substituting these and the G'EANT service is being re-designed with SDN applications run on the controller. This is an ongoing project which intends to evolve the current network and schedule services over time and to manage the network dynamically. Even though they have had failures, the applications are running with defects. If we had standard strategies to perform such a migration to the next generation, we would have had several NRENs shifting to SDN which would result in much-improved productivity in research and education, specifically in Africa, as open-source proves to be a key component of addressing legacy network infrastructure problems. This chapter further looks at available techniques of undertaking the migration of SDN in the WAN use case with the aim to accelerate the deployment to be applied or implemented in educational or institutional networks in developing economies.

2.3 Network Overlay

Numerous experiments on network overlay have been carried out to tackle the switch to SDN. (Corici et al., 2016) presented a SDN-Overlay architecture which was integrating the factory environment network architecture at high impact sites with SDN deployment. The proposed technique enables the data path full adaptability if any system malfunction happens. The researchers suggest that somehow the transfer of controller activities within the network system should always be granted watchful capabilities to mitigate the bottleneck transmitted throughout the OpenFlow Network. The research was carried in order to assess the proposed controller versus Ryu and Floodlight but the new controller did not handle network traffic better than Ryu and Floodlight. Yet only centralized open-source controllers had been used and could not be treated as decentralized controllers for a WAN environment. In fact, the analysis revealed no criterion for choosing such controllers. Nonetheless, the researchers suggest that the most viable structure for further deployment inside the factory setting is an SDN dependent adaptability system. Furthermore, given the ultimate goal of implementing these configurations as a part of a distributed system, the allocation of controller devices in the system should be carefully considered. This research consequently embraces a distribution methodology for using SDN-Overlay.

In WAN it is necessary to position the correct number of SDN controllers, and this was investigated by (Mamushiane, Lysko and Mwangama, 2018) in which researchers diagnosed the issue of controller positioning in SDN in the sense of the developing WAN driven environments, respectively SANReN and ZAMReN. The researchers introduced a numerical model to tackle the problem of controller placement with special emphasis on delay in transmission. The study indicate that operating a sole controller induces a relatively high delay in transmission due to the fact that certain switches are placed farther distant from the controller, which is why the alternative of positioning SDN agents at just the boundary of a SANReN point of presence (PoP). Nevertheless, the amount of controllers to still be configured relies on the unique capabilities and shortcomings from each supplier. The findings indicate that using two controllers was necessary to deliver improved QoS performance. This research adopts the method of deploying multiple SDN devices for the WAN environment.

Mijumbi et al. (2016) suggested an alternative for improving the interaction of highly diverse overlays. A solution is applied in the light of ANBS to monitor overlays to improve their performance. The overlaying device is defined as the SDN controller, the residual overlay points are substituted through as network switches, and thus the control layer is separated from data layer. Each controller gathers and manages overlay content, and sometimes moves the data to the database system. The results indicate how the proposed strategy has enhanced heterogeneous overlay efficiently. Our research holds a similar path by taking the OpenFlow devices nearer to the conventional network boundary. (Wang et al., 2017) Implemented Cluster Based Network Partition Algorithm (CNPA). Knowing that the amount of deployed network equipment from each subnet will change in response to regional allocations, the researchers recommend that multiple controllers be deployed to minimize idling latency arising from unnecessary packet queries from switches. Given the final goal of evaluating the efficiency of the proposed solution, experiments were performed using the Internet Topology Zoo against two actual configurations. The findings indicate that CNPA will adequately reduce the most severe end-to-end delay amongst controllers and the corresponding control devices. CNPA has also significantly minimized overall delay compared to K-means and K-center in a cycle where multiple controllers were deployed in subnets. A similar strategy can be used to incorporate various controllers within demarcated SDN overlays.

2.4 Switch Placement

It is crucial to position the correct number of controllers in WAN. This was discussed by Mamushiane, Mwangama and Lysko (2018), where they defined the issue of controller placement in SDN in the context of developing economies, centred on WANs, in particular, the South African National Research and Education Network (SANReN) (Wara and

Singh, 2015) and the Zambia National Research and Education Network (ZAMReN) (Chembe et al., 2014). The researchers proposed a numerical model specifically focused on propagation latency to address the problem of controller placement. The results revealed that operating a single controller created a higher average lag in propagation since some switches were placed farther away from the controller, which was the alternative to position SDN devices on the edge of a SANReN point-of-presence (PoP) was developed. The number of controllers to be used depends on each service provider's particular requirements and restrictions. The analysis has shown that QoS performed best when two controllers are used. This study, therefore, has triggered the need to assess the influence of the data plane devices on the overall network performance and further implement techniques to improve the performance of both traffic engineering and network management.

Huang and Liang (2017) studied the SDN switch configuration issue and suggested a heuristic methodology by eliminating traditional switches algorithmically, in order to optimize overall network performance. The researchers formulated a numerical algorithm for scalability optimization that substituted several switches for each execution rather than just a sole switch. The researchers suggested a plan that would replace current traditional switches in a wide-scale infrastructure to optimize the effect of the new SDN switches. The approach suggested indicated an increase in network scalability, (Guo et al., 2014) and (Levin et al., 2014) also considered systematic deployment of SDN switches to address the issue of placing the SDN devices through iterative methods.

Hamasuna et al. (2017) conducted a study on cluster validity measures (such as Dunn's Index, Davies-Bouldin's Index, Fuzzy Covariance Matrix Based Indices, etc.) for clustering network data. In their simulation experiments, DI illustrated improved outcomes than most other cluster validation tests. Experimentation revealed that the development of new cluster validation mechanisms centred around the Dunn Index concept, which were numerically established, resulted to new clustering techniques for network systems. Wang et al. (2017) implemented network partitioning along several subnets using a streamlined K-means to reduce the gap throughout the subnet between both the controller and the SDN switches. The findings of that same simulation revealed that perhaps the overall latency reached by the alternative modified K-means algorithm is much reduced than that of the regular K-means.

Kuang et al. (2018) implemented an algorithm centred on a hierarchical K-means for resolving the issue of controller positioning. Using only a hierarchical technique, the researchers subdivided a huge network up into multiple different SDN cluster nodes. The researchers chose the network configuration from Internet2 OS3E to assess the proposed algorithm. Results of the experiment indicated that somehow

the algorithm presented was much more robust than that of the modified K-mean algorithm. The configured K-means indicated improved scalability than that of the algorithm presented by the researchers, while it did not demonstrate improved load balance relative to the solution presented.

Zhou et al. (2016) presented a methodology for estimating the optimum number of clusters generated by the AHC algorithm has been introduced. The researchers evaluated the potential index for multiple datasets including linear, multiple, annular, and convex data. The researchers opted for AHC since it provided relatively less time for complexity and increased operational stability. The recommended index computed the optimum amount of correlations for the datasets under analysis and the researchers indicated that it can be used with wider datasets. Yet they did not equate the conceptual and experimental outcomes to those of other validation assessments to establish a concrete evidence of the effectiveness.

2.5 Load Balancing

On the data plane, hybrid SDN consists of SDN and conventional devices that exist side by side in the network, but it is managed through the conventional devices distributed in the control plane. An SDN switch is aware of the legacy protocols that require traffic management by techniques such as load balancing. Load balancing is a method of managing external network traffic by objectively assigning and transferring traffic loads among reachable network resources to improve network accessibility, reduce latency and the bandwidth usage (Bhatt et al., 2018). A load balancer is a tool designed on custom hardware in traditional networks, but load balancers are costly to maintain and manufacturer-specific (Zarin and Agarwal, 2018). For SDN, load balancers are such a software which can be controlled to resiliently monitor and control the processing of network activities. Controllers including ONOS, Ryu, Pox and floodlight, include static network load balancers with defined load balancing principles (Nkosi et al., 2018). Load balancing is commonly applied throughout the SDN controller through two types of balancing, i.e. stateless and stateful load balancing (Pit-Claudel et al., 2018) and (Ganesh et al., 2017). In stateless balancing, a controller does not have network status tracking abilities and it can only balance network traffic at a custom-defined time. For stateful, the controller tracks the condition of the network and helps to balance the network as needed. Although the pure SDN method has mostly been favoured, it is wasteful as the controller has to keep all flow states on the flow table. (Zhou et al., 2017).

Recent research on the data plane layer by Nkosi et al. (2018) assessed the dynamic performance of multi-path load balancing on a fully OpenFlow-based SDN data plane for wide and small-scale networks. The evaluation included only two SDN network

topologies and measured the efficiency of the ODL load balancer compared to that of the multi-path load balancer. The load balancer improved network performance in terms of transfer rate and response time. [Zakia and Yedder \(2017\)](#) introduced a technique similar to an SDN-based adaptive load-balancing algorithm ([Nkosi et al., 2018](#)) for data centre networks. To automate the use of network links, the classifier discovers the shortest links for each host to other nodes and calculates the cost of the connection, thus trying to take the flow priority into consideration. If congestion occurs, this technique replaces the old path with the alternative route with the lowest connection cost. The studies conducted by ([Nkosi et al., 2018](#)) and ([Hu et al., 2018](#)) used the shortest route technique and the lowest link cost method by using the Dijkstra algorithm to calculate the shortest path for each flow and balance the network among alternative paths. This research seeks to use a similar method and assess the developed load balancer under various topologies in a hybrid environment.

The implementation of SD-WAN should not be perceived only in a theoretical sense, as it is thought to be complex since two different architectures are involved. ([Xu et al., 2018](#)) designed a hybrid network switching methodology on a testbed, and incorporated conventional switching and SDN switching in order to achieve interoperability and optimum performance respectively. To run the evaluations, the researchers used the ODL open-source SDN controller. In fact, to reduce the routing rules, a hybrid route delivery approach was introduced. Load-balancing re-routing was also suggested as a case study for implementation. Test and statistical analysis tests showed that hybrid switching performed better than the complete SDN network architecture. The convergence process also resulted in an unintended benefit of the hybrid design as it rendered all forms of switches more effective. In order to achieve both scalability and optimal efficiency, [Xu et al. \(2018\)](#) introduced a hybrid switching system on a test platform by combining conventional switching and SDN switching. Furthermore, the researchers introduced a technique of hybrid path deployment to minimize the rules for routing packets. Load-balancing re-routing was also suggested as a research study for application

[Bhatt et al. \(2018\)](#) introduced and evaluated load-balancing techniques, including Randomized Hydrodynamic Load Balancing, Cogset Load Balancing, Entity Resolution Block Balancing, Ant Colony Optimization and Shortest Path. The researchers placed the methodologies into Hadoop and then compared the techniques theoretically. Efficiency was examined throughout the map phase, delay time, response time, performance, turnaround time and threshold and reduction phase. The researchers found that in at least one parametric function, each algorithm had good outcomes, but the Cogset Algorithm provided the most resilient outcomes and improved performance compared to other algorithms. Cogset, however, is static, not flexible, based on data redundancy.

2.5.1 Economic Analysis

While SDN offers a handful of benefits, this technology also faces corporate and financial obstacles such as budget constraints, appropriateness and relationship building among telecom operators, which has an impact on SDN's adoption for full implementation. The shift of market-related problems can therefore only be a concern for service providers who provide end-users with a seamless operation. Consequently, SD-WAN and MPLS can coexist and this can reduce the costs related to MPLS networks, as SD-WAN is capable of exploiting both internet and MPLS traffic engineering capabilities to offer business functions. Emerging markets are often described as developing economies or developing countries as detailed in Figure [2.1](#), which are shifting investments to alternative productive capacity ([Santacreu, 2015](#)) through the migration from traditional economies that are dependent on agriculture and foreign trade of raw materials to industrializing and embracing a free market or mixed economy.

Volatility Index from 1 (Lowest) to 5 (Highest)

Country	2012	2013	Change
Brazil	2.50	2.67	0.17
China	2.17	2.33	0.16
India	4.17	4.00	-0.17
Indonesia	2.67	3.00	0.33
Malaysia	3.00	3.33	0.33
Mexico	2.67	2.83	0.16
Poland	2.83	2.50	-0.33
Russia	1.67	2.50	0.83
South Africa	3.33	3.33	0.00
Thailand	2.83	2.67	-0.16
Turkey	3.33	3.50	0.17

NOTE: Calculated as a simple average of ratings based on each country's short-term debt, foreign reserves, current account, inflation, cyclically adjusted public budget, and government debt.

SOURCE: International Monetary Fund, Haver Analytics, the central bank of each country, and the author's calculations.

Figure 2.1: Volatility Index for Emerging Markets ([Santacreu, 2015](#))

Organisations frequently replace existing equipment or acquire new equipment to develop business functions. Both decisions necessitate the organisation to assess the future cash flows to choose whether or not the investment will boost the value of the organisation ([Gitman et al., 2015](#)). The studied financial techniques are used by senior management to assess investments that implicate financial forecasting for long-term projects that can extend over a year or more is called capital budgeting ([Gitman et al., 2015](#)). When it comes to asset acquisition and replacement there is often a substantial threat associated with future financial uncertainties such as theft, incorrect equipment and malfunction of equipment resulting in frequent maintenance. Hence, it is important to take capital budgeting decisions into account when adopting a networking migration plan.

Dawadi et al. (2018) highlight the lack of funds, skilled professionals and most importantly appropriate migration plans as major challenges for service provider networks. The authors present a greedy technique for cost identification in migrating network devices based on customer demands on SDN-enabled IPv6 networks. The observation in (Dawadi et al., 2018) is that joint migration is a solution for reducing CAPEX and OPEX on service provider networks. But the joint approach becomes an additional challenge for NRENs because this would require at least an African network such as ZAMREN to work with SANREN. Likewise, Mamushiane, Lysko and Dlamini (2018a) propose infrastructure sharing for mobile networks, specifically in emerging markets, to reduce CAPEX and OPEX. When a financial model is used to assess the impact of shared infrastructure, the results display the positive change in the profitability period, therefore reduced costs for customers, small start-ups and savings for larger operators. This paper by (Mamushiane, Lysko and Dlamini, 2018a) quantifies an optimal technique by using a financial model through hybrid SDN, but NRENs pose a different challenge when it comes to network sharing, as some would be structured as non-profit and others are government-driven while most of them are operating independently. Therefore, we explore a more refined model that considers the time value of money and various financial factors.

Karakus and Durresi (2018) conducted an economic analysis of different popular SDN control plane structural designs. The analysis involved the economic influence on the service unit costs of control plane structural designs for different traffic patterns, the total cost of ownership and network revenue for different network owners. The authors categorized the unit cost of service in accordance with the network CAPEX, OPEX, and workload for a specific time and applied mathematical methods to different SDN control plane architectures. The experimentations and examination of the acquired results revealed that the Centralized Control Plane (CCP) model showed the best Total Cost of Ownership (TCO) whereas the DCP LV model showed the smallest amount.

In another study by (Karakus and Durresi, 2019), numerical analysis for CAPEX and OPEX calculations of a network, the work evaluated unit service cost scalability and cost to-service. However, NRENs were not for commercial use and the models presented were tailored for developed markets, and the emerging market case was not considered. Therefore, using the cost of service as a measure of economic implications and considering its case as a commercial enterprise became a challenge.

Gitman et al. (2015) suggest that capital budgeting is important when assessing a financial plan for a long term project that requires asset acquisition or replacement as it takes into account all the future cash flows which are derived from investments at a targeted weighted average cost of capital. Often there are unexpected associated

monetary implications as organisations commit to present known resources and future unknown resources that involve several uncertainties. The results suggest that project estimation and cash flow assessment have been identified as the most significant feature of the capital budgeting process. Authors ([Gitman et al., 2015](#)) further suggested that more consideration should be given to project assessment as small companies were more exposed to the possibility of insufficient financial recovery.

The hybrid SDN migration technique promises a number of advantages through enabling SDN features when connected to the traditional network, such as central control of the network, which is combined with the advantages of traditional architecture such as inexpensive deployment budgets and time-evaluated-maturity ([Wara and Singh, 2015](#)). Thus, we get the best of both worlds. Hybrid SDN offers the capabilities to revert to matured and tested legacy appliances in case of unexpected network failures during the transition, which is not possible in a completely new pure SDN deployment ([Lee et al., 2018](#)). Updating or installing multiple rules on a device that is centralized can be problematic in entirely SDN network setups. This could be the result of channel clogging, congestion, the capacity of an SDN controller to process the rules. However, with the use of both centralized and distributed-control capabilities in a single network environment, these above-mentioned issues can be addressed. When congestion experienced in communication resulting in the inability of the SDN controller to respond to traffic loads, one can take advantage of the legacy switch-distributed routing capabilities to route operative packets ([Vissicchio et al., 2017](#)). Through the provision of central control over direct traffic, controller overhead is reduced and controller scalability is maximized.

Table [2.4](#) shows related work on network overlay, switch placement and network load balancing with their shortcomings. The solutions to overcome these drawbacks will be presented by this work.

Table 2.4: Literature Gaps Summary

Literature	Contribution	Shortcomings
<u>Corici et al. (2016)</u>	The study provided adequate device route adaptability will there be a machine malfunction event. Reducing the congestion supplied throughout the OpenFlow Network.	The research only used non-distributed controllers, never including dispersed controllers that might be deemed suitable for a WAN configuration and the analysis revealed no selection criteria for these controllers
<u>Mamushiane et al. (2019)</u>	Found the issue of controller positioning in developing economies, that concentrated on WANs, including SANREN and ZAMREN.	The amount of controllers to install depends upon each operator's specific needs and restrictions.
<u>Wang et al. (2017)</u>	Suggested to set several controllers to reduce the idling delay arising from unnecessary requests via switches.	The proposed technique is compared to relatively old clustering techniques while there are improved versions of those techniques.
<u>Cui et al. (2016)</u>	Suggested an approach to boost the association across diverse overlays.	The use of centralized controllers and no consideration of large network scenarios.
<u>Huang and Liang (2017), Guo et al. (2014) and Levin et al. (2014)</u>	Studied the SDN-enabled switch placement problem aiming to make the most of system performance, they proposed Optimization algorithms by the iterative replacement of traditional switches.	The incremental procedure is time-consuming and costly since it involves many trial and error processes and does not place the SDN switches optimally on the network.
<u>Zhou et al. (2017)</u>	To address the issue of load amplitude among several controllers and improve network stability, a workload-balancing approach was suggested for the data plane based on the switches category.	The study did not consider WANs and heterogeneous networks to test the proposed method.
<u>Bhatt et al. (2018)</u>	Introduced and evaluated load balancing techniques, including Randomized Hydrodynamic Load Balancing, Cogset Load Balancing, Entity Resolution Load Balancing, Ant Colony Optimization and Shortest Path	The most popular technique, that was Cogset, is rigid, and not dynamic, relying on data repetition.
<u>Karakus and Durresi (2018)</u>	The study included the impact on the economy of aircraft command system structures on service cost per unit under different operating conditions, full ownership costs and network profits for different service owners.	NRENs are not for commercial use and the models presented are tailored for developed markets and the emerging market case is not considered

The shortcomings presented in Table 2.4 will be addressed by means of experimentation and evaluations in Chapters 4,5 and 6. Finally, the relevant conclusion to the study is presented in Chapter 8, which highlights how each objective was dealt with and resolved.

2.6 Chapter Summary

This chapter presented the literature of this study. The literature also highlights the transition or the migration to SDN in the WAN environment. But moving to SDN presented challenges in previous years, when researchers dealt with controller placement challenges and with that have come the switch placement challenges which are still a hot topic in the SDN space. On top of that, there are emerging markets from the first world playing the catch-up game. Researchers have seen and started to explore hybrid SDN as a possible solution when it is deployed in WAN. T presents a better solution for emerging markets as the financial challenges also play a vital role in the delay in the advancement of technology. The related literature presents the baseline for conducting this study, and Table 2 shows how previous work has contributed to the research space and the shortcomings. The next chapter outlines the methodology this study will use to achieve its objectives. The methodology will provide detailed approaches and techniques together with resources that will be used in the study.

Chapter 3

Methodology

3.1 Preamble

This chapter discusses the research methods and the procedures involved in the development of the study. This study follows a theoretical philosophical approach to focus on going towards programmable networks in a practical way. The theoretical and practical assumptions are taken into account in the evaluation of the SD-WAN ([Eyisi, 2016](#)). This study will gather and analyse both primary and secondary data.

3.2 Applied Methodologies

Figure [3.1](#) illustrates the structure of the research methodology used in this work, where the qualitative analysis will provide descriptive (who, what, where, when, why and how of the research) ([Nassaji, 2015](#)) and explanatory analysis (clarifying concepts, gathering explanations, gaining insight, eliminating impractical ideas and forming hypotheses) ([Jajoo, 2014](#)) on the technological ideologies or theoretical perspective of the applicability of the SDN technology in WAN infrastructure through a hybrid SDN approach. This information will eventually serve as a basis for the analysis and conclusion of the investigated migration strategy in this study.

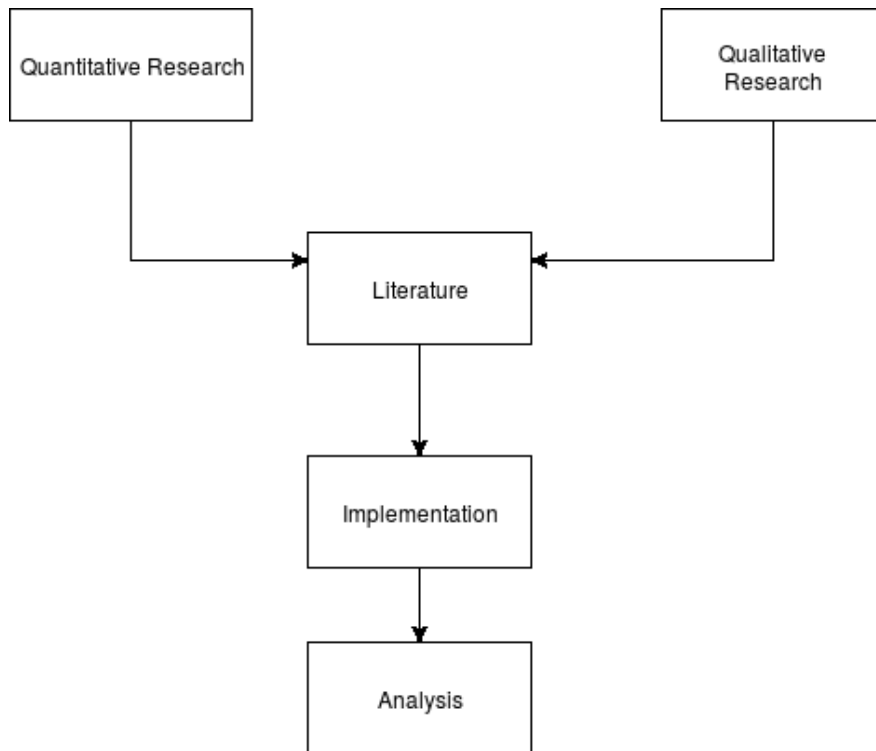


Figure 3.1: Methodology Framework.

Figure 3.1 illustrates the top-down method followed in this study. The study details the two types of research approaches which form research and related work stage that takes into account a number research sources that assist in building an overall foundation for this research study and shows how past researchers have dealt with the problem under study and highlights the research gaps that still exist in the research space. This will assist in formulating a problem and identifying the required tools (these may be simulation or emulation tools, mathematical models to solve the problem or a testbed if necessary). Once the literature has been gathered and analysed, we will move to the implementation step. Using the information from the literature (datasets and qualitative and quantitative analyses) we can identify the problem, find possible techniques to solve the problem, then also find the relevant tools and other necessary resources to solve the issues at hand, such as the Cbench simulation tool (*cbench*, n.d.) and Mininet (Imt, 2017). Upon achieving our results we will move to the analysis step, where we analyse the results by using graphical visualizations and numerical analysis to draw the overall conclusion based on the results obtained.

The main intent of the Literature and review of standards is to enlighten the reader about the work that has been done and to show how far these technologies have shaped the road of SDN and what can be achieved with the knowledge we have to date, this will also leave a clear understanding of the knowledge gap affecting the

implementation of the SDN infrastructure. The quantitative analysis will draw practical and objective conclusions about the primary and secondary results obtained.

3.2.1 Quantitative Research

The quantitative research in this work will be used to define the problem and producing numerical data that can be converted to practical statistics (Jones, 2004). This approach will involve gathering data relevant to WAN data sources, research publications and network experimentation results. The study is offering a detailed analysis of benchmarking methods and resources for SDN controllers. It covers the current research & methods utilized for measurement, the nature of the benchmarking techniques and, most notably, the specifics of the measures to be used for predictive evaluations.

Evaluation or benchmarking of the performance of the controller through simulation or emulation. While hardware test beds have metrics that are similar to real values in the production environment, their costs are high for the research community. Evaluations focused on simulation are also standard practice. Nonetheless, the analysis framework used for benchmarking SDN controllers is effective and reliable. The benchmarking will be an independent test setup and should not be connected to devices that may forward the test traffic into a production network or misroute traffic to the test management network. Further, benchmarking is performed on a single machine basis, relying solely on measurements observable external to the controller.

3.2.2 Qualitative Research

Qualitative Research is mainly investigative research. The data collected will be processed by means of technological explanations. It will also be categorized into numerous segments and subsets. Different types of figures will also be used to provide a more detailed and concept-oriented view of the problem. Then it will be analysed and explained, typically by means of technology examination. A small amount of quantitative data will also be used in this work to present applicable details for the research.

Study will discuss the classification requirements for controllers for their properties and capabilities, accompanied by a comparative review, and then various use-specific changes. The study also discuss the various use cases for these controllers as well as the improvements produced to better their efficiency.

3.3 Implementation

The implementation process involves the experimentation of a wide variety of mathematical, theoretical and practical methods. The experiments will be repeated in similar environmental setups to safeguard the uniformity and correctness of the results achieved. The setups mentioned involve software technology and an SDN lab environment. Virtual networks (VNs) are beneficial in several applications such as training or experimenting with different functionalities which reduce the cost barrier as there is almost no need for physical infrastructure. With the advancement of this principle of network slicing, different virtual operators can independently use the same tools. The implementation will involve simulation through virtual machines of SDN and legacy networks (as hybrid networks) using tools such as Cbench and Mininet.

Two types of analysis will be done on the controllers; one will be performed on the flow setup, and the other on the latency of the flow setup. These two main measurement parameters have been used to determine controller performance. The emulated switches transmit as many PACKET IN messages as feasible to the controller inflow configuration throughput mode, meaning that perhaps the controller has messages to handle at all times. The flow configuration latency experiment uses Cbench to mimic switches that transmit a single packet to the controller, listens for a response, and afterwards repeats the process as rapidly as possible. The total number of responses received by the controller at the end of the cycle is used to measure the controller's average time to process every PACKET IN triggered by the switch.

The following criteria for measuring both latency and bandwidth are considered for the OpenFlow Controller's benchmarking methodology:

- OpenFlow switch count: the total number of OpenFlow switches to connect to the controller.
- The total number of hosts (Flows): hosts linked by the controller.
- OpenFlow Version: the version of the OpenFlow protocol used by the controller to connect.
- Test Iterations: it is important to perform the test several times.
- Test length: test iteration period, measured in seconds.

3.3.1 Simulation Tools

Cbench is a program for evaluating the performance of an OpenFlow-compatible controller by stressing it with Packet-In messages. It emulates OpenFlow switches which connect to a controller, push Packet-In messages and waits for Packet-Out messages, whilst measuring various metrics. This tool mainly operates in two modes namely, throughput and latency.

In the current project, we found it more practical to model this physical network by using Mininet. Its aim is to provide the basic infrastructure that will establish the layer of virtualization/emulation. To do so, we define topologies which match the needs of the study without adding excessive complexity. Such topologies can be defined by using different techniques; we have opted to describe them in our set-up by using Python's Mininet libraries. One can insert commands such as the number of switches, hosts, topology sort, etc. However, in the end, it becomes simpler to compile them into a Python script. Figure 3.2 illustrates the list of simulation tools used for the experiments.

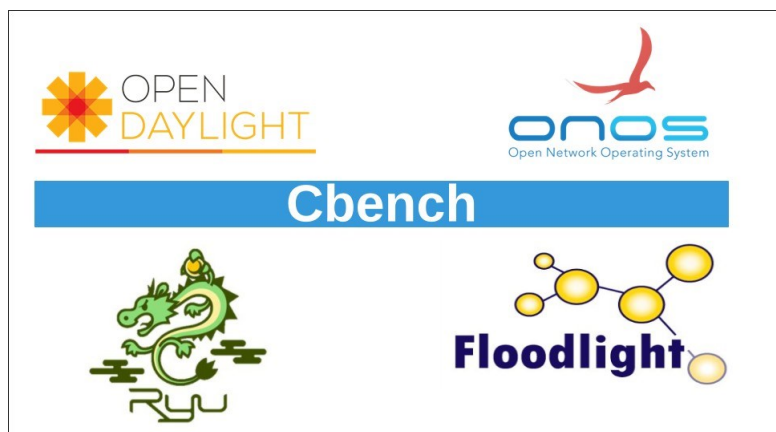


Figure 3.2: Simulation tools

3.3.2 Test Environment and Tool Setup

The testing process was carried out on OpenFlow simulation tools. The controllers and the benchmarking tool were installed on the same host in order to overcome the Ethernet interface speed limitations. We ran multiple parallel instances of both the benchmarking tool and the controllers during the experiment to use all available resources. The results of the first two iterations will be overlooked for controller warm-up purposes.

Setup and Test: Overlay

To overcome the Ethernet interface speed limitations, both Cbench and the controller under test (Floodlight, Ryu, ONOS and OpenDayLight) were implemented under a shared Ubuntu server 16.04 LTS-64. The Cbench and Mininet tools were set up by using a single PC (Dell Latitude 5580), with an Intel Core i7 vPro 7th Generation processor. All the controllers were in their isolated VMs.

Figure 3.3 illustrates the setup of our experiment. Cbench is used to emulate the data plane. The experiment is divided into two parts. First, to investigate the impact of the control load on controller latency (the time it takes to receive a Packet-Out message after evoking a Packet-In message). Secondly, the same evaluation is done to measure the impact on throughput.

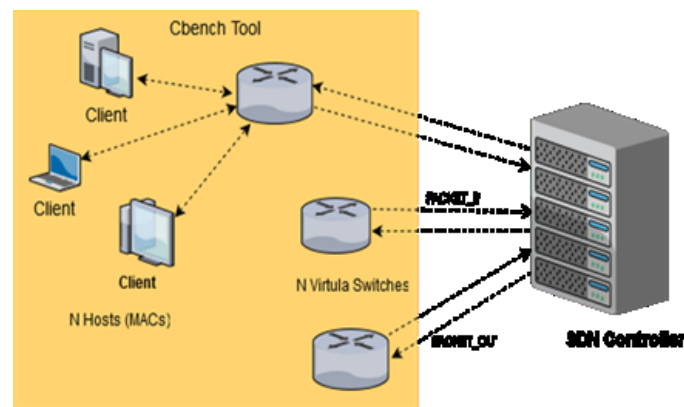


Figure 3.3: Setup of the benchmarking experiment

In the first part of the experiment, the number of switches is varied from 1 to 4 and then from 4 to 32 in increments of 4. Each time, the switches are connected to the controller under test using OpenFlow and they send Packet-In messages to the controller. Cbench's statistic and performance probe modules are used, the Packet-Out messages are counted, and the latency is measured. Here, the number of MACs (a shorter version of media access control address) identifiable as hosts is fixed at 1000 MACs while the number of switches is varied. This experiment is repeated with varying numbers of MACs (1K, 10K, 100K, 1000K, 10000K) while the number of switches is fixed at 16.

The second part of the experiment is performed to measure throughput by varying the number of switches and MACs in the same manner as in the first experiment. The number of worker threads was set at 8 to maintain a fairly low runtime. To reduce the influence of time-sharing in a virtual environment, 14 iterations per test are executed in both experiments. The duration of each test is set at 10 seconds.

The first two loops are considered the controller warm-up and their results are ignored. The following example command is used to carry out the test:

```
./cbench -c localhost -p 6633 -l 14 -m 10000 -M 1000 -s 8 -t
```

Where

- *c*: controller IP or hostname (localhost);
- *p*: controller port number (6633);
- *l*: number of iterations (14);
- *m*: test duration in ms (10000);
- *M*: number of MACs per switch (1000);
- *s*: number of switches (8);
- *t*: throughput mode.

Setup and Test: Placement

The first test evaluated the two algorithms by performing a numerical evaluation. The studies were performed on a 3.20GHz x 12, Core i7 CPU 40 GB ram, 1 TB hard disk and Ubuntu 16.04 operating system cloud-based server.. To maintain realism, the optimization solution was applied to a real-world national backbone called SANReN (a South African national backbone). SANReN is a WAN distributed across seven cities in South Africa. The major factor in the mathematical model is distance, whereas the delay throughout all domains is unchanged. Thus transmission delay is directly related to distance under constant bandwidth.

Setup and Test: Load Balancing

The Dijkstra algorithm will be used in the multi-path load balancing approach to determine the shortest alternate path of each flow and to optimize traffic between alternative flows ([Kim et al., 2017a](#); [Nkosi et al., 2018](#)). The load balancer efficiency is measured employing four different topologies, namely, topology for the fat tree, ring mesh and torus. The experiment incorporates Mininet to mimic topologies, whilst also Wireshark tracks throughput, and Iperf to produce traffic loads.

Setup and Test: Economical Analysis

SANREN provides 10 GB links to connect research and education institutes but on a normal utilization basis, only 1% to 50% is utilized, therefore, with 47 sites, core routers are connected to 300 campuses of 85 institutions. In 2017 alone TENET billed institutions an amount of over R145m (SANReN CA, 2018). Among the network equipment such as border routers, core routers, switches, backup power and servers, the switches account for an estimated amount of 20% of the overall capital expenditure on traditional network equipment (Foley, 2016). First, the study conducts a project cost estimate to predict the cost of developing and deploying a full SDN setup using the hybrid approach (Chan et al., 2008). The associated cost estimate and project activities will be used to analyse the capital budgeting of the hybrid SDN deployment in the space of African NRENs.

3.4 Analysis

The analysis will include tables and graphs that analyse the performance of SDN controllers in hybrid design architecture and the application scenario of each evaluation. Subsequently, when data collected from the research part is used, an objective conclusion can be drawn and analyzed. A technology comparison will also be part of the analysis. It will serve as a discussion of the benefits of each of the solutions examined.

3.5 Chapter Summary

In this chapter, we have discussed the research methods that will be used in this work to conduct experiments and analyse the results achieved and generated. The main intent is to provide an indication of how the research is organized and the approaches that will be followed throughout the study. First, the structure of the method is outlined in Figure 3.1 in the form of a diagram to show how the relevant steps will be followed. This section presents the details of the research approach each subsequent chapter will follow. The implementation phase of the methodology starts in Chapter 4, where the study attempts to expedite the changeover to SDN by implementing the SDN overlay alternative for a conventional WAN environment, and by targeting the provision of flexibility and control in the event of network failure.

Chapter 4

Hybrid SD-WAN Overlay

4.1 Preamble

The adoption of SDN in WAN offers an efficient and price-effective telecommunications framework with diverse infrastructure and application process traffic engineering. Implementation of existing Wide Area-Networks (WAN) SDN systems and implementations takes longer than in data centre networks. Nonetheless the use technology such as SDN overlay, and having more SDN-enabled equipment potentially shorten adoption times, especially when management is enhanced in WANs leveraging interoperability of control planes and network management, which would have a major impact on the development of wide-ranging networks. Figure [4.2](#) shows an overlay SDN switch solution that can be placed on top of the existing traditional network. This research aims to promote the migration to SDN by overlaying SDN while preserving traditional services. The main focus for the new conventional WAN setting is on an SDN-driven overlay approach to assess the performance of the SDN controllers and target the provision of flexibility and autonomy over a decentralized SDN infrastructure overlay and edge SDN equipment in the case of network disruptions.

4.1.1 Assumptions

The following assumptions apply to the proposed solution:

- **Out-of-band switch-to-controller communication:** Links are used to interchange control data between both the controller and the forwarding devices controlled by the controller or between controller instances.
- **Constant bandwidth connection to all links:** The propagation time interval is unchanged.
- **Load to all switches is fixed**
- **Switches and controllers are co-located.**

4.2 Proposed Solution Overview

The architecture presented offers a SDN overlay approach for the WAN framework in emerging markets as it implements distributed control by co-locating each enabled SDN with each point-of-presence (PoP) and enabling distributed controllers across the WAN to minimize delay even though the scale and composition of the metropolitan areas can differ. The implementation seeks to provide consistency and control of the data route via a decentralized overlay of both the SDN network as well as to edge SDN devices if any technical failures arise. The design splits the wider WAN apart into metros, for this experimentation, it is the SANREN as shown in Figure 4.1. The above essentially allows controllers throughout the metro infrastructure and deploys SDN equipment (OpenFlow devices) at the periphery of the conventional network such that the overlay traffic is separated from the data plane to minimize latency.

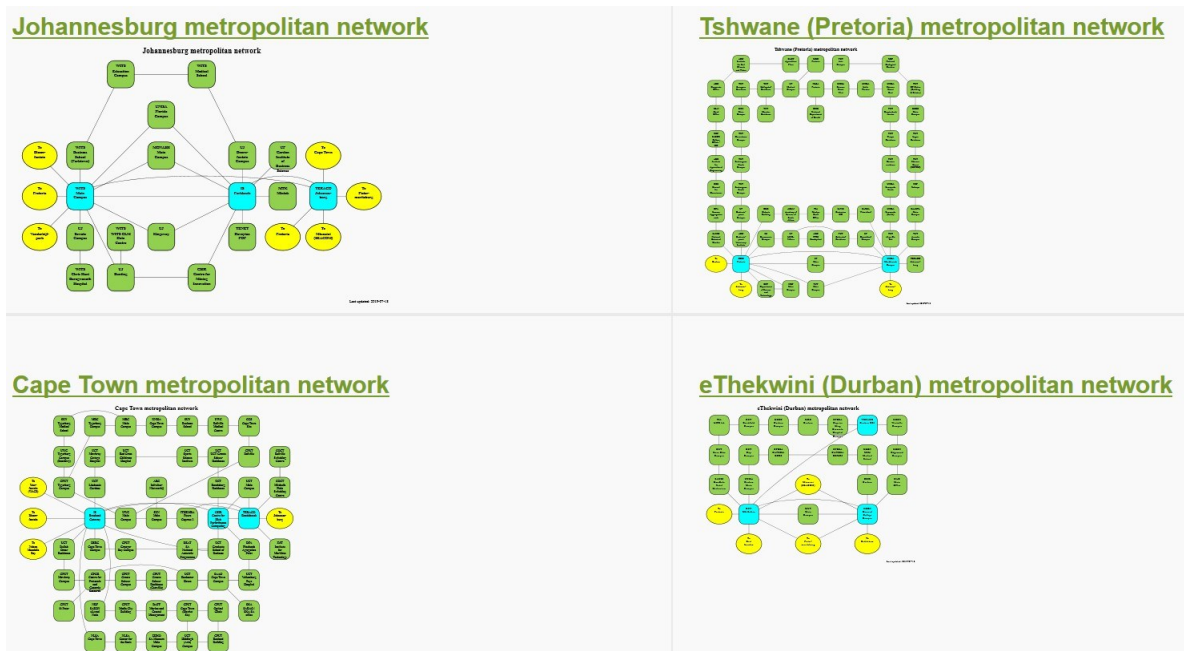


Figure 4.1: SANReN Metros ([SANReN CA, 2018](#))

The SANReN in Figure 4.1 comprises nine operating metros and two under planning. It is constructed to guarantee the effective contribution of South African researchers and educational institutions. The SANReN is operated by the CSIR's Next Generation Enterprise Enablement and Institution and is also part of the National Integrated Cyberinfrastructure System (NICIS), along with the Centre for High-Performance Computing (CHPC) and the Data-Intensive Research Initiative of South Africa (DIRISA). The network is connected to the UbuntuNet Alliance ([Alliance, 2014](#)), the regional Research and Education Networking organization for Eastern and Southern Africa. SANReN has provided the network device information of their WAN for this study.

The virtualization is performed as of the periphery of the infrastructure, however what exists of the L2/L3 infrastructure remains intact and does not involve any shift in configuration to support the network virtualization. The proposed overlay acts as a middleman utilizing techniques like those of VXLAN, these can act as both the virtual tunnel endpoint (VTEP) or network virtualization edge (NVE) ([Mamushiane, Lysko and Mwangama, 2018](#)), STT and NVGRE. And in the most sections, overlay-riven approaches can also be applied over legacy systems with little or no adaptations being mandatory ([Goransson et al., 2016](#)). The controller establishes an unified device topology state and handles data flow decisions onto the acceptable forwarding route, depending on the base station (BS) queries obtained., Floodlight, Ryu, OpenDayLight and ONOS. To achieve this, a benchmarking tool called Cbench ([cbench, n.d.](#)) is used.

Figure 4.2 reflects a software-defined SANReN Johannesburg Metropolitan infrastructure, in which a configuration of SDN controllers (C) is positioned within two location points which are also connected to several other metropolitan stations. The data traffic is exchanged over an array of switches (SW) to the controller. Also, to have the capacity to exploit the shortest path the network traffic is transmitted to a controller through a series of switches (SW), The SDN devices will be able to send information to base stations. Thus managing the heterogeneous devices of the conventional and SDN network is much less difficult for an overlay.

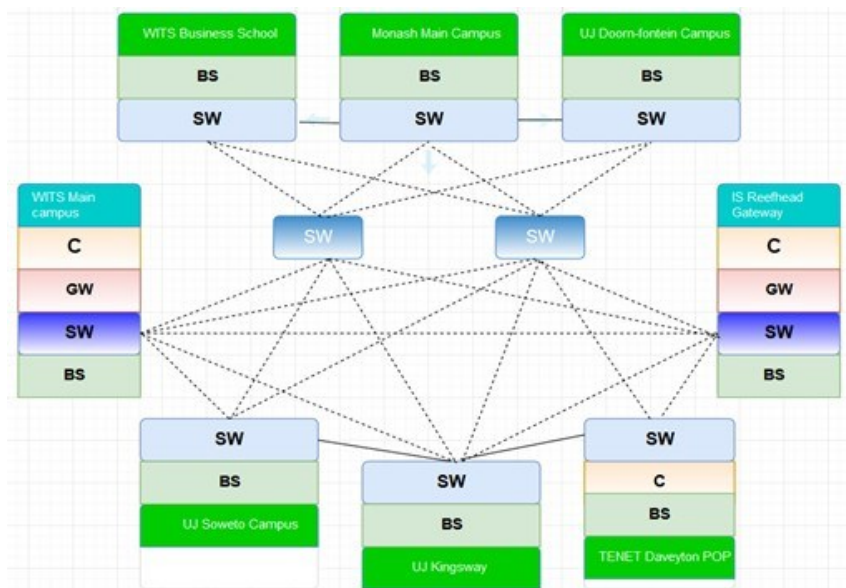


Figure 4.2: SDN Overlay WAN Architecture

An overlay infrastructure is layered on top of a conventional network. These are facilitated by overlaid data messages encapsulated in several other network packets. IP packets will be stored up and transmitted to the desired destination. Due to a data packet being just a set of bits, the elegance of this method lies in the ability of a controller to be placed on the edge and direct the traffic. The advantage of the network overlay is the fact that the traffic passes on top of the physical network whilst the hypervisors transmit and receive data from the overlay. The overlay network has knowledge of the data packets at the endpoints which is transmitted through physical devices. Nonetheless, the endpoints are completely clueless about the particulars of the network topology, the networking procedure or any other form of standard or intermediate network operations. It helps us to distinguish the data flow clearly from the packet control often because network packets are a collection of bits, these bits can be modified to mimic what the movement of packets across distinct physical connected devices.

4.3 Flexibility Evaluation

The flexibility is analysed in terms of control plane distribution as a function in which a network can evolve or adapt to changing flow routes or change of the allocated resource over a given time. The evaluation is conducted by using the topology design that is illustrated in Figure 4.2

4.3.1 Controller throughput

As shown in Figure 4.3, there is a significant difference in the throughput exhibited by various controllers under a varying number of switches. It is clear that Ryu's throughput performance is poorest. This can be attributed to the fact that Ryu is resource-intensive and uses CPU and RAM optimally which results in performance degradation in the presence of an increasing number of switches (Asadollahi et al., 2018).

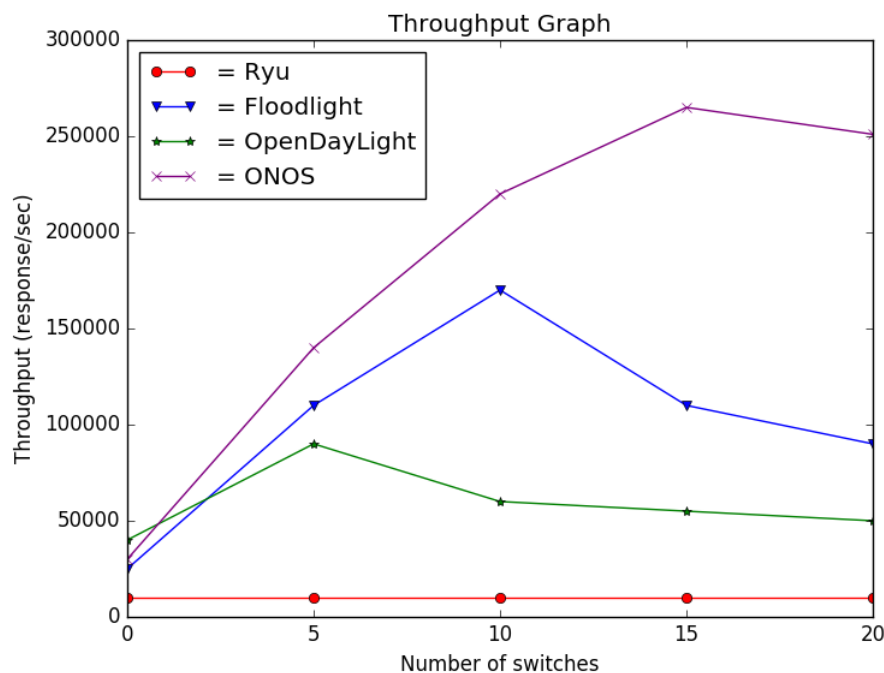


Figure 4.3: Throughput vs Switch Graph

OpenDayLight and Floodlight yield a low throughput when the number of switches is increased beyond five and ten switches, for OpenDayLight and Floodlight respectively. As with Ryu, these controllers are more resource-intensive and require activation of hyper-threading to improve performance. ONOS shows the best throughput

performance, probably because it is multi-threaded and adds more switches, which leads to better CPU utilization.

4.3.2 Controller scalability

Figure 4.4 shows the results achieved when the number of MACs per switch is varied under a fixed number of switches. The results indicate that OpenDayLight and Ryu throughput remains virtually unchanged under a varying number of MACs. Therefore, these controllers are not ideal to use for large-scale SDN deployments. Floodlight shows a high throughput for a smaller number of MACs (up to 10 000 MACs) and shows the diminishing performance when the number of MACs is set beyond 10 000.

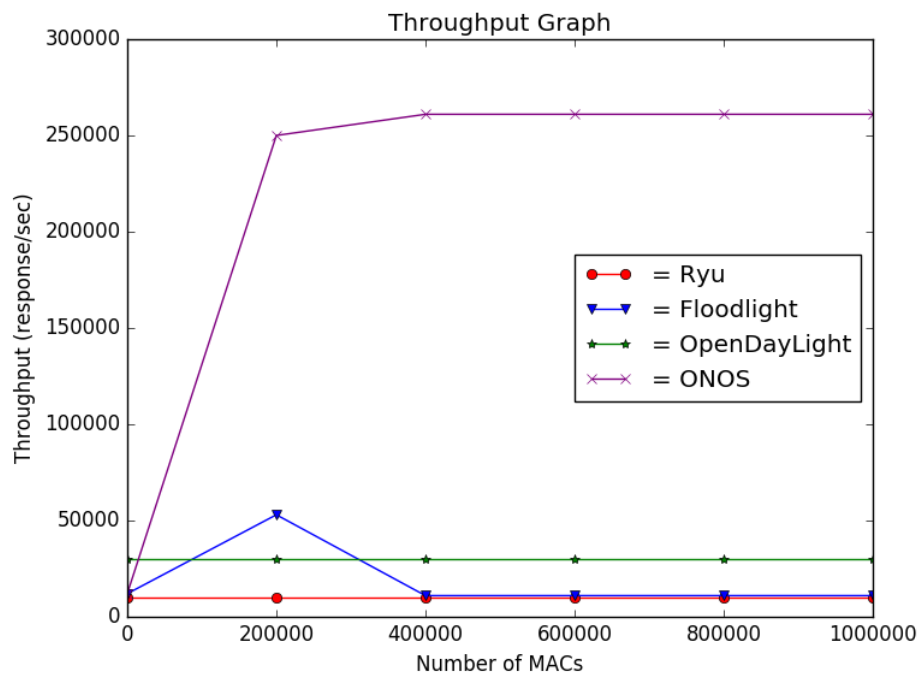


Figure 4.4: Throughput vs MACs Graph

The ONOS controller in Figure 4.4 has the best throughput performance compared to that of its rivals. This is probably due to the ONOS data plane contention management module that splits the MAC address table among a collection of hash tables, which results in improvement of the ONOS responsiveness (Lam et al., 2018). OpendayLight showed an interesting performance where it remained constant even when the number of MACs was increased. This can be explained by the ODLs MAC learning through the use of a learn action, to minimize large-scale punting of packets to the controller for an unlearned source MAC.

4.3.3 Controller latency

The results achieved in the throughput mode are shown in Figure 4.5 and Figure 4.6. The results show that ONOS yields the worst latency performance both under a varying number of switches (as shown in Figure 4.5) and a varying number of MACs (as shown in Figure 4.6). Ryu stands out, as it shows the best latency performance. The latency remains low and constant under a varying number of switches. This is probably because Ryu offers limited support for multi-threading. OpenDayLight's performance is better than that of Floodlight under a varying number of switches.

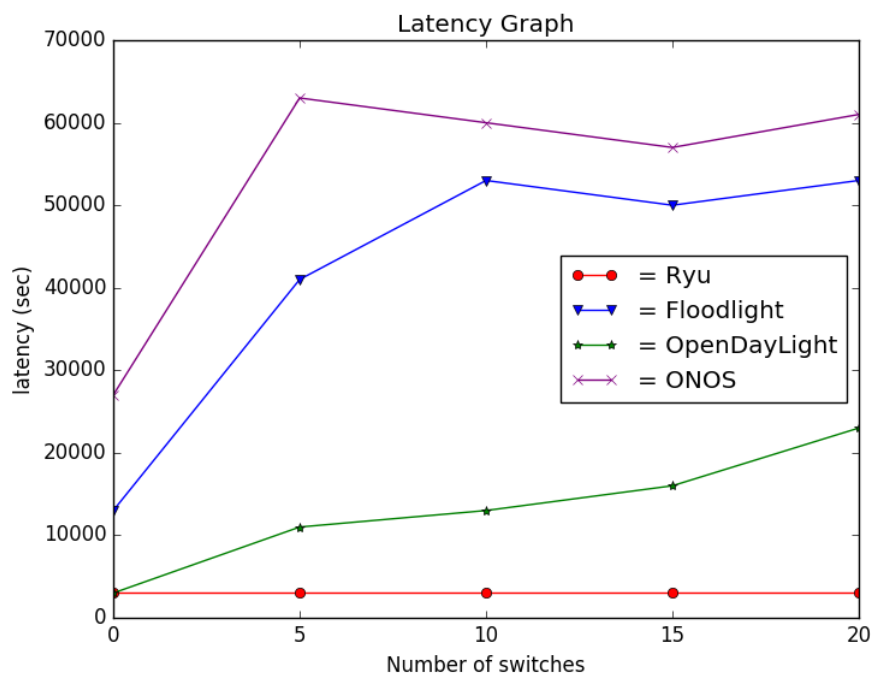


Figure 4.5: Average latency under switches

The latency results in Figure 4.5 shows the Ryu as the optimal controller when the MACs count increases over time. Although Floodlight and OpenDayLight are also displaying improved performance compared to ONOS with the increase of MACs.

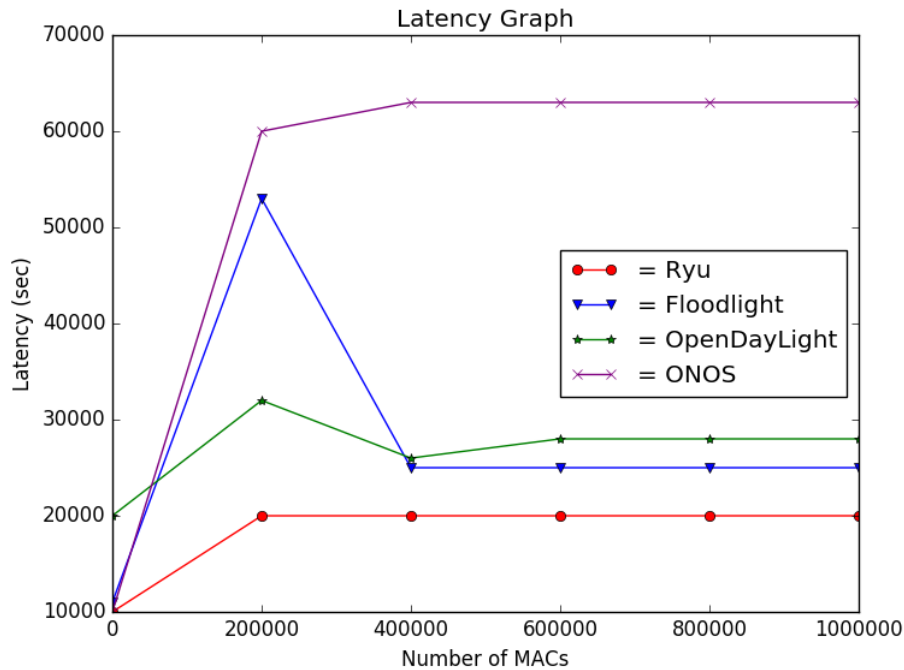


Figure 4.6: Average latency under MACs

When the number of MACs is increased beyond 100 000, OpenDayLight’s performance becomes significantly poorer than that of Floodlight, which reveals its scalability limitations. Therefore, networks using Floodlight require more time to find the route and send instructions for new flows.

4.4 Reliability Evaluation

An experimental evaluation using the Mininet emulation framework was performed throughout this section that included transmitting ping commands on a specific topology route. A request sent messages to a virtual host, then reported the Round-Trip Time (RTT) and measuring bandwidth utilization ([Mamushiane, Lysko and Dlamini, 2018a](#)). The Iperf data transfer measurement method was implemented to provide an estimation of the network capacity for the chosen metropolitan topology, and the most intense transfer rate attainable. From the prior versatility test the ONOS controller was picked because it demonstrated greater stability characteristics both in delay and bandwidth utilization scenario scenarios.

Figure 4.2 illustrates the Mininet-simulated topology. To prove the edge SDN-node's advantage, we acknowledge the delay in propagation, it takes time for a signal switch to transmit from one node to another (Shozi et al., 2019). The following formula can be used to calculate delay;

$$d_{prop} = \frac{D}{s} \quad (4.1)$$

Where, D is the distance between the node and the node below, and s is the media propagation rate. The transmission delay is calculated to be equivalent to $3.3\mu\text{sec}$ in free space and $5\mu\text{sec}$ in fibre for a single packet that crosses a connection at 1m/sec . For every 1 Gbps sent to the SANReN link

$$P = 1 \text{ Kbyte}$$

$$R = 1 \text{ Gbps}$$

$$100 \text{ KM, fiber}$$

$$P/R = 8 \mu\text{sec}$$

Where P refers to the number of bits R refers to the bandwidth in bps, the propagation time is P/R . Delay of propagation = $T = \text{duration}/\text{speed}$. With Figure 4.7 and Figure 4.8, ten pings were sent, and no packet drop was observed with the configuration comprising three controllers. Ping test 1 is H1 ping H2, Test 2 is H1 ping H3 and Test 3 is H1 ping H5.

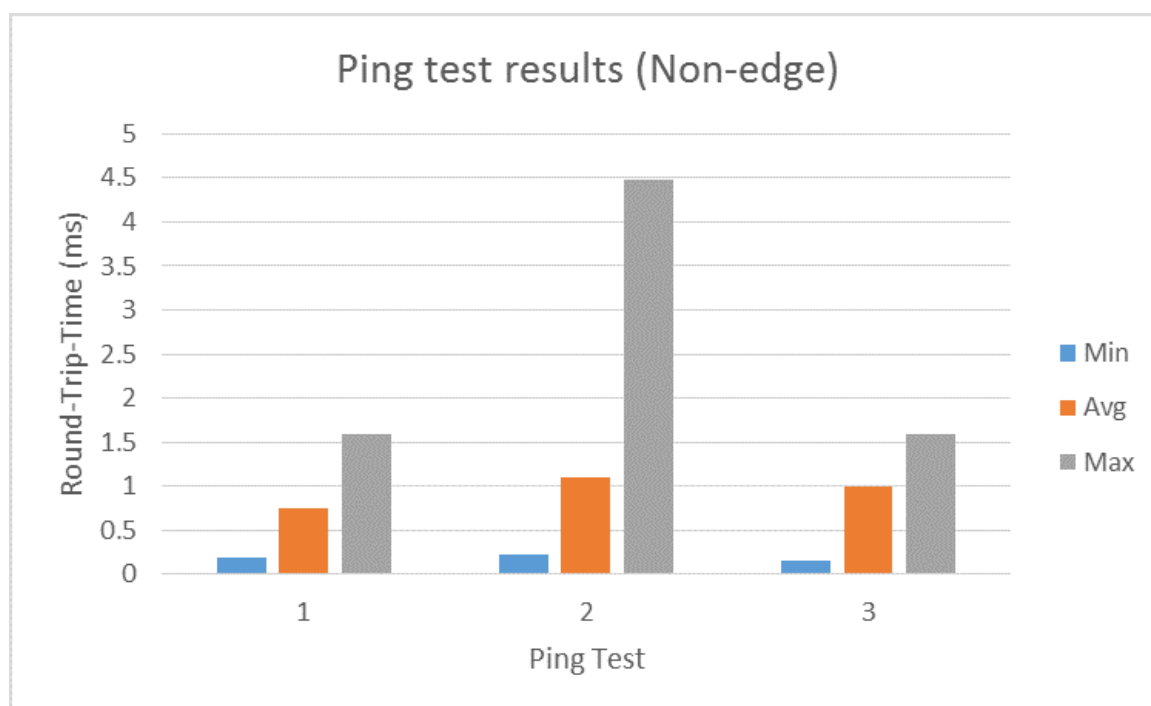


Figure 4.7: Ping test results (Non-edge)

An mean round trip time (RTT) is relatively low whenever the SDN nodes are close to the boundary of the network opposed to situations in which the nodes are farther from the edge. The steady mean RTT with edge configuration in Figure 4.8 shows that ONOS doesn't really adjust its value over time.

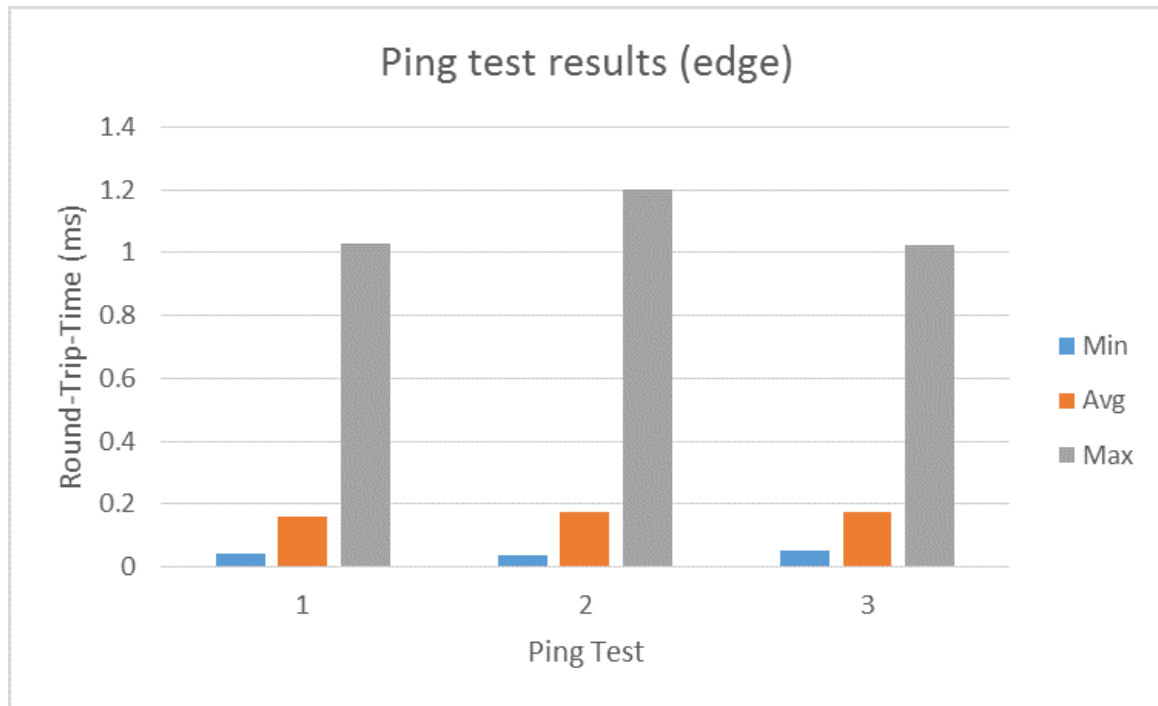


Figure 4.8: Ping test results (Edge)

The spike of the maximum RTT in Figure 4.8 opened up some questions for further research in network overlays but edge devices showed improved performance in all the scenarios that were tested. Placing the overlay devices on the edge of the WAN did reduce the network delay. The RTT in Figures 4.9 and 4.10 use of the two-controller system was repeatedly tested. Ten pings have been sent, and there has been no decline in the amount of packets throughout. Ping test 1 describes H1 ping H2, Test 2 was H1 ping H3 and Test 3 was H1 ping H5.

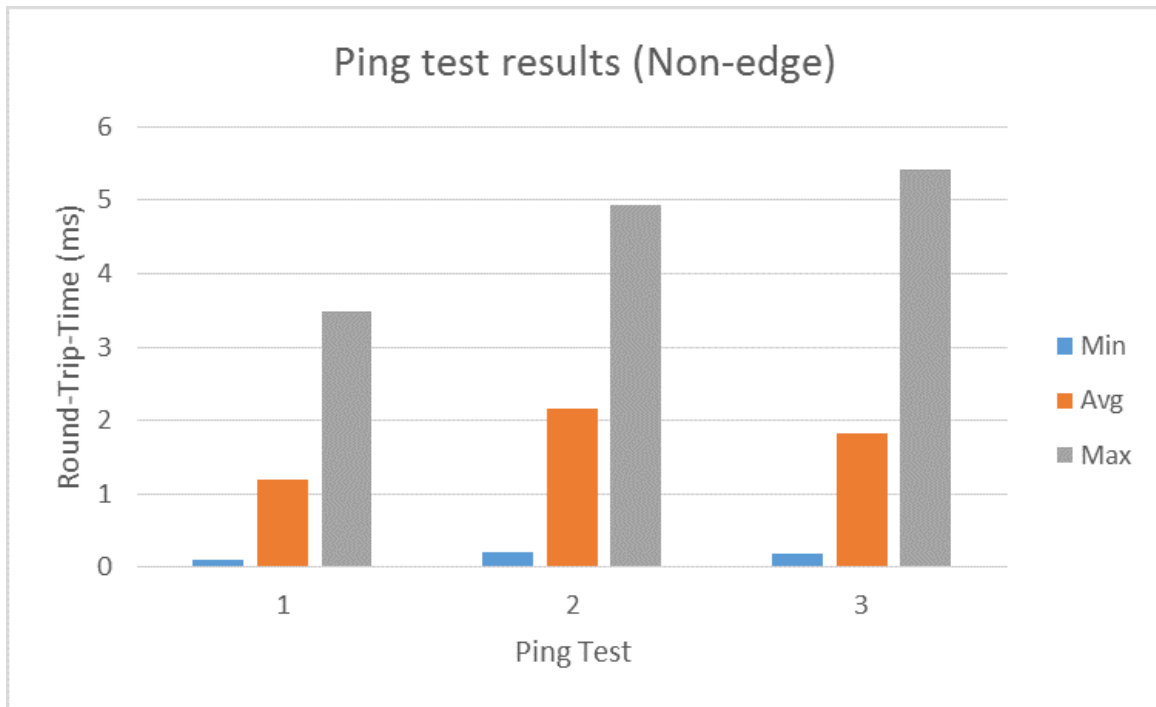


Figure 4.9: Ping test results (Non-edge)

Again, the SDN-edge nodes in Figure 4.10 proven to have been more productive than those of non-edge devices, with comparatively higher RTT from such a lowest average of 1.182 to 2.166 ms.

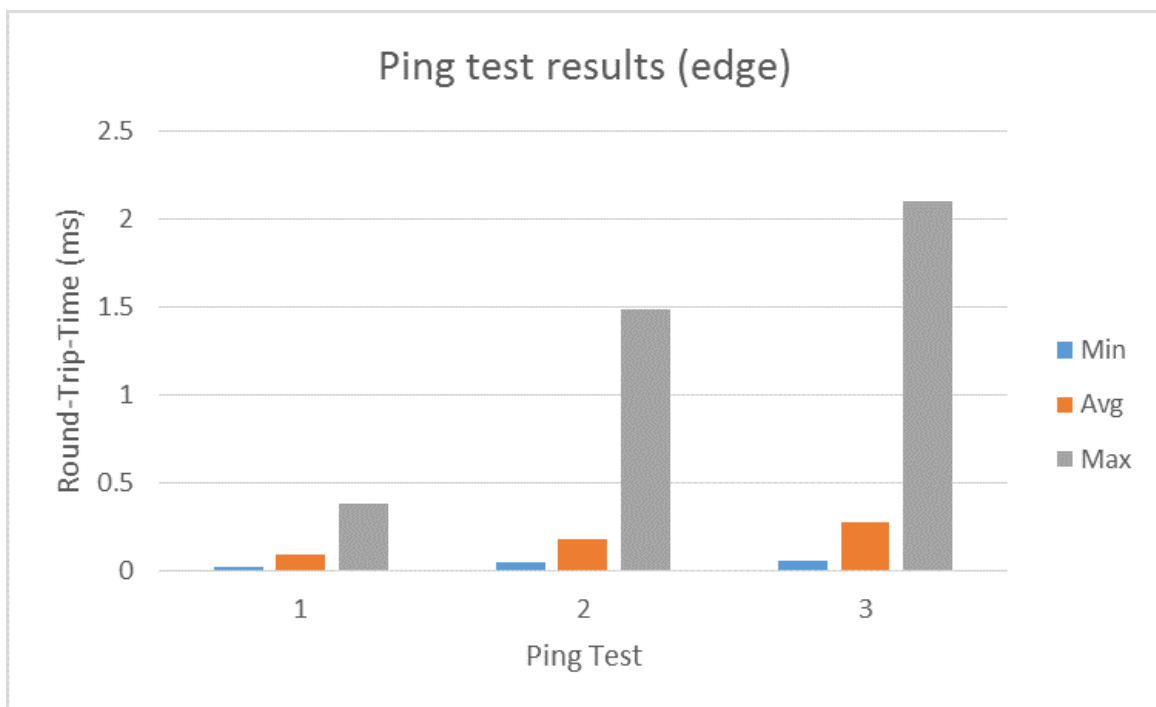


Figure 4.10: Ping test results (Edge)

The two-controller setup proves to be a better setup than that with three controllers on the network. Observation of the edge device results which have shown a positive reduction in latency proves that the maximum of Figure 4.10 is relatively more optimal than that of Figure 4.9 and also that the average RTT of the two-controller setups is less than that of three controllers. The results of the number of controllers that is optimal for the emerging market use-case of a WAN when placement is taken into consideration are similar to those achieved with the work on controller placement in the African backbone (Mamushiane, Lysko and Mwangama, 2018). Another interesting analysis of the results of non-edge devices shows that the average RTT is better when three controllers are deployed rather than two controllers. This shows the necessity of switch placement in WAN because if the switches are not placed optimally, an organization can end up using an additional controller which will result in increased CAPEX and OPEX. As NRENs in Africa suffer from insufficient funds to support network operations, using this technique can save them a lot of money.

The TCP throughput was tested in edge and non-edge environments both for the two and three controller setups, and thus the outcomes are seen in the Figure 4.11 in the non-edge configuration, the two-controller configuration of edge devices requires additional bandwidth (up to 51.6 Gbps) compared to non-edge devices, as they are not affected by latency during packet transmission.

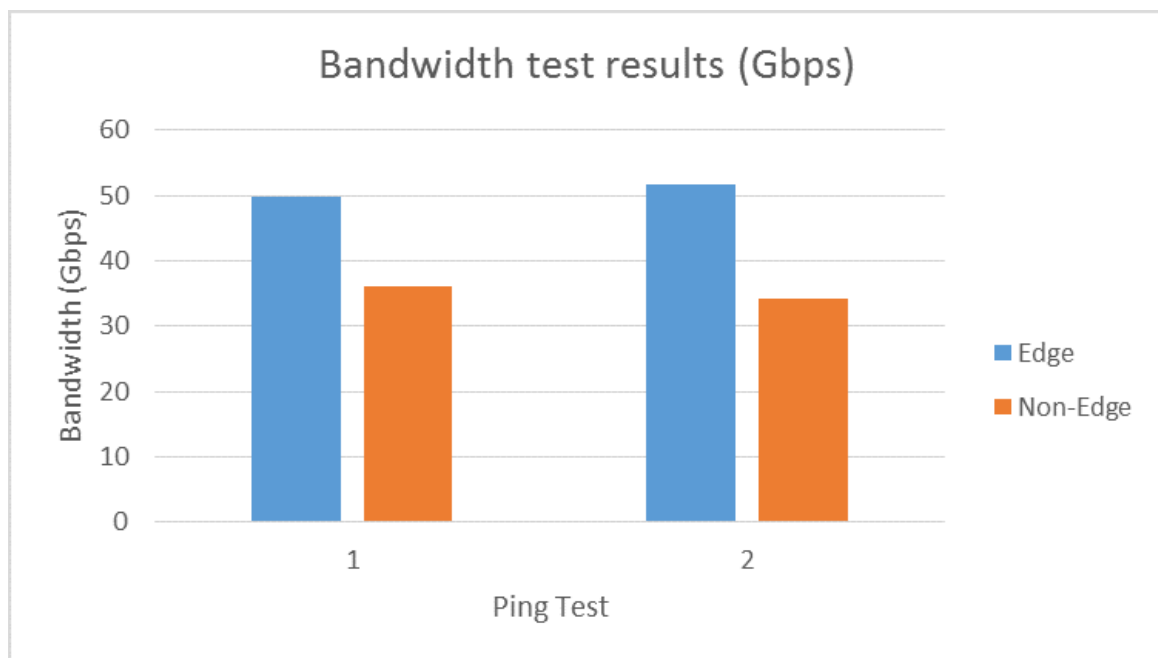


Figure 4.11: Bandwidth test for edge and non-edge placement

In view of the evaluation of the bandwidth utilization, it is safe to note that the edge placement has once again shown performance improvement compared to non-edge devices.

But what is also notable is the number of SDN controllers. The first ping test had three SDN controllers and the second had two controllers. Judging from the edge placement alone the two-controller setup again performed better than the three-controller setup, with a 1,6 Gbps difference.

4.5 Chapter Summary

This chapter has addressed the first and second objectives in Section 1.5 by presenting the alternative for current conventional WAN infrastructure is via a decentralized SDN overlay environment as well as edge SDN nodes aimed at providing versatility and management in the occurrence of network failures. The ONOS outscored its competitors in the versatility evaluation, by delivering the best bandwidth utilization when managing the increased amount of network devices and MACs. There had been a meaningful change throughout the RTT in the ONOS performance assessment, as three controllers have been used in lieu of two, with overlay points on the boundaries. The overall RTT was decreased to very little than 0.2ms and, very significantly, the centralized management constraint was reduced the bandwidth utilization of edge devices was also improved. This will allow network engineers to plan better and optimize the network for peak performance in order to be prepared for network failure and possibly avoid such failures.

With respect to decentralized management, edge overlay device positioning and latency, a flexibility structure based on SDN overlays is practical for any further configurations within WAN. The contribution of this section was focused on finding an optimal deployment solution for the transition to SDN, where it proposed a solution that provided flexibility and facilitated management by reducing delay and utilising bandwidth efficiently. This work has also assessed open-source controllers to find a suitable controller for a hybrid SDN overlay and we have contributed regarding the controller scalability issue by showing that the two-controller setup is more optimal than the three-controller setup.

The placement of switches on the edge does not completely suffice for the deployment of hybrid networks or completely solve the switch placement problem without the optimal placement of the switches on the network. Literature also suggests that SDN switch placement will provide useful and productive traffic engineering, subsequently improving network performance ([Chen et al., 2017](#)). Other data plane deployment techniques initiate supplementary hardware abstraction equipment such as hybrid upgrade architecture that may affect network performance negatively. ([Casey and](#)

Mullins, 2015). While solving the deployment problems, it is important to use optimization techniques (which also affect the order of deployment of SDN devices) to improve the usage of limited SDN switches in hybrid SDN networks. In Section 5, we extend the study by looking at the placement of SDN switches in a hybrid network and proposing a placement algorithm to place SDN switches optimally.

Chapter 5

Hybrid SD-WAN Switch Placement

5.1 Preamble

SDN enables more effective distribution of bandwidth capacity by modifying interactively to adjusting requirements and handling failures faster. However, considering the next generation of Wide Area Networks (WANs), which are required to be simple, versatile, reliable and affordable, the complete transition to SDN infrastructure will be challenging. One of the challenges is the fact that full deployment of SDN, whereby all nodes on the network are SDN enabled, cannot be accomplished directly in the short term. Therefore, a possible and efficient solution is to shift to the adoption of SDN hybrid networks while retaining legacy infrastructure. In hybrid SDN networks, the placement of multiple SDN controllers and the limitation of the number of SDN-enabled switches on the hybrid network are still challenging, owing to issues with network operation stability, and financial constraints. In this chapter, we adopt the idea of network partition to solve the SDN-enabled switch placement challenge in hybrid SD-WAN. In order to address this challenge, we explore the Agglomerative Hierarchical Clustering (AHC) algorithm and the Optimized K- means to solve the SD-WAN partitioning and switch placement problem and use a cluster validity measure to determine the number of optimal clusters.

Previous studies on controller placement ([Mamushiane, Lysko and Mwangama, 2018](#)) and edge node placement ([Shozi et al., 2019](#)) in the SANREN as a use case for African NRENs have been explored as a strategy for moving towards the adoption of the SDN paradigm, but there is a lack of a strategic methodology or technique in placing SDN-enabled switches on a large-scale network and, specifically, hybrid SDN networks. The placement of SDN switches will lead to convenient and effective traffic engineering, which will result in network performance improvements ([Chen et al., 2017](#)). A limited number of conventional switches can be substituted by the SDN-enabled switches, in certain optimal domains and positions.

This is commonly referred to as a hybrid SDN solution. An important question to be answered during transitioning to a hybrid solution is: Given a legacy WAN, how many SDN switches are required and what is the best locations to place them so that the overall network performance is optimized? ([Amin et al., 2018](#)). Henceforth, we will refer to these questions as the switch placement problem.

5.1.1 Assumptions

The development of the mathematical method is based on the following assumption:

- All the nodes on the network share the same physical link with 10 Gbps;

5.2 Algorithm(s)

5.2.1 Agglomerative Hierarchy Clustering

Agglomerative clustering is a popular type of hierarchical clustering which is often chosen to group objects in clusters based on their similarity. Agglomerative procedures start with n singleton clusters and then form a sequence by successively merging clusters. Considering that the data set is x_1, x_2, \dots, x_n and G_i is the i th cluster in the k th merging process ([Abe et al., 2017](#)). The AHC pseudocode is as described in Algorithm 1 below. The AHC begins with every node in a single cluster. Then, in each continuous iteration, it merges (agglomerates) the neighbouring pair of clusters by satisfying similarity criteria. Until all the data is in one cluster, smaller clusters are created, which are useful in network discovery.

The minimum distance between clusters serves as a measure to find the right number of clusters on the network, where the maximum value of the index represents the right partition with the highest separation between clusters and less spread data in-between clusters. The devices with the maximum range to the centroid and the modified centroids are chosen specifically as that of the original centers within the next partition. The process repeats before subnets are subdivided into k .

Algorithm 1 Agglomerative Hierarchy clustering

```
1: A set of  $X$  objects  $\{d_1, \dots, d_n\}$ 
2: A distance function  $\{dist(c_1, \dots, c_n)\}$ 
3: for  $i = 1$  to  $n$  do
4:    $c_i = \{x_i\}$ 
5: end for
6:  $C = \{c_1, \dots, c_n\}$ 
7:  $I = n + 1$ 
8: while  $C.size > 1$  do
9:    $(c_{min1}, c_{min2}) = \text{minimum } dist(c_i, c_j \text{ for all } c_i, c_j \text{ in } C$ 
10:  remove  $c_{min1}$ , and  $c_{min2}$  from  $C$ 
11:  add  $\{c_{min1}, c_{min2}\}$  to  $C$ 
12:   $I = I + 1$ 
13: end while
```

Algorithm 1 starts from a set of discrete entities that are first clustered into small clusters; these are in turn clustered into larger clusters until they are finally inclusive clusters (Ozaki et al., 2017). One advantage of these algorithms is the fact that they are non-supervised.

5.2.2 Optimized K-means

The optimized K-means is derived from the revised K-means algorithm (Wang et al., 2016) which is also developed from the standard K-means algorithm (Wang et al., 2017). The revised version differs from the standard K-means in three important steps: Step 1, centres are initialized, and the interpretation is similar to that in Algorithm 2. Step 2, corresponding to smallest route distance, the devices are allocated into clusters. And step 3, Corresponding to the devices within a cluster, the centroid is modified. Such phases establish the third step of the algorithm optimized for K-means.

The fundamental of the updated K-means approach was the incremental spread of a full network from 1 to K subnets. As abridged in Algorithm 2, calculating the quickest route is the very first stage of optimized K-means. After that a point is chosen randomly from V to declare the modified K-means (Wang et al., 2017). Since the subnet G only has one centroid (the network has not been subdivided), irrespective as to where the original center is located, Algorithm 1 searches for a system centroid in Phase 3. In phase 4, the algorithm must consider the subnet goal which would be subdivided into the next portion.

The intended subnet is discovered by measuring the total range between both the centroid and the subnet points. The original centers will be initialised in phase 5. The node with the longest range to the centroid and the modified centroids are chosen specifically as that of the original centers in the next partition in step 6. Steps 4, 5, and 6 are replicated till the subnets are divided into K .

Algorithm 2 Optimized K-means network partition algorithm

- 1: Compute the shortest paths $d(u, v)$
 - 2: Randomly select one node from V as the first initial centre ($centre(1)$) of G ;
 - 3: Do the revised K-means algorithm and find out the exact centroid of $G(centroid(1))$.
 - 4: Find out the target subgraph (G_{target}) which has the greatest distance (shortest path distance) between its centroids and nodes. The node that has the greatest distance to the centroid is assigned to the centre (G_{target})
 - 5: Select the previous centroid (or centroids) and the $centre(G_{target})$ as the new initial centre
 - 6: Do the revised K-means algorithm again and find the centroid of each subnetwork
 - 7: Repeat steps 4, 5 and 6 until the network is partitioned into K subnetworks
-

5.2.3 Dunn's Index

Dun's Index (DI) is considered one of the necessary cluster validity tests for the assessment of the network traffic node partitions (Hamasuna et al., 2017). The methodology facilitates in estimating the number clusters needed as it operates with cluster density by concentrating on maximum distance among data points and minimal distance among clusters (Bhadana and Singh, 2017). DI has been used to evaluate partition quality and the "finest" crisp subdivision of n entities is then identified (Wang et al., 2017).

5.3 Algorithm Evaluation

We initially evaluated the complexity of data-processing time using the Optimized K-means against the AHC using the xclara data set that consists of 3000 nodes. The xclara (Hayashi et al., 2017) is a widely used data set in statistical evaluation and consists of a large dataset compared to that of SANReN, therefore it will be suitable to assess data-processing time. Figure 5.1 shows the elbow method used to determine the k value for the K-means. Thereafter, the performance of the proposed approach with the integrated validity measure alongside the Optimized K-means will be evaluated in terms of latency and throughput by using realistic network data.

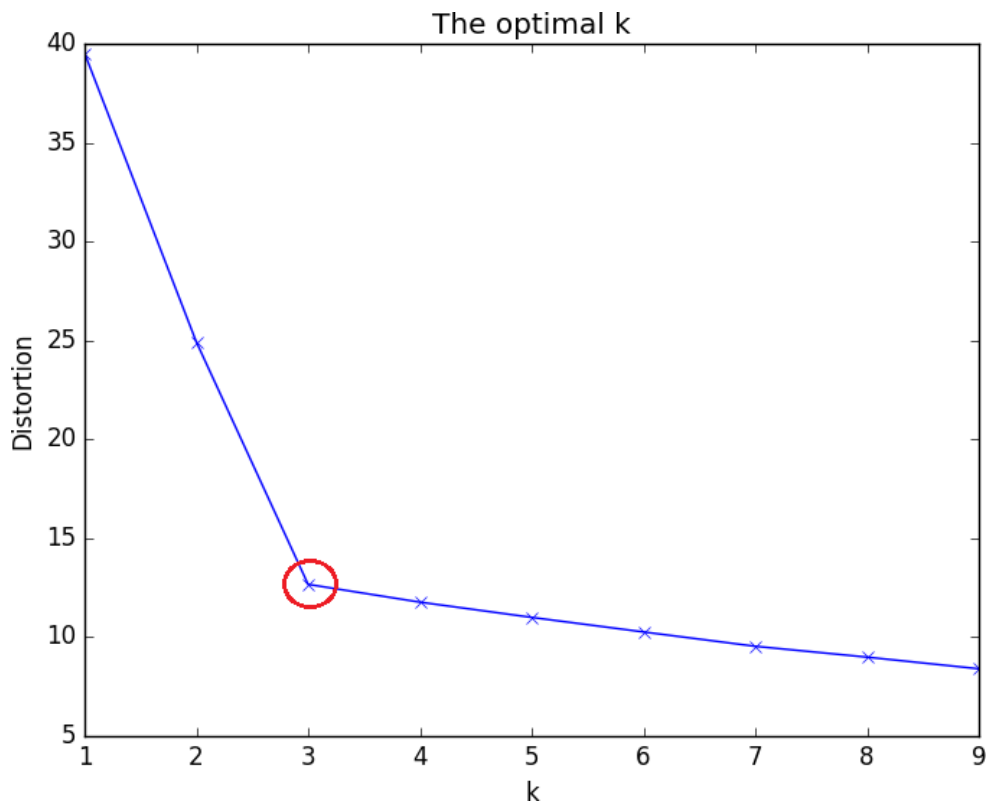


Figure 5.1: Validity Index Graph.

Figure 5.1 indicates the k value (circled in the graph above as value 3) which is calculated by using the K-means elbow method to find the value of k that we need before evaluation of the complexity of the data processing time.

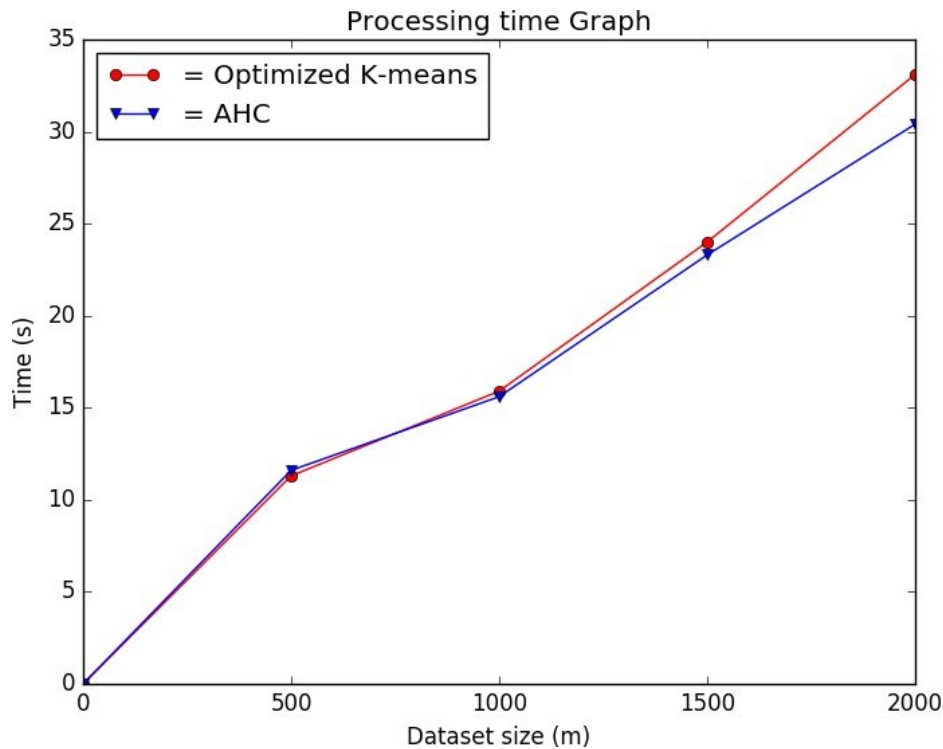


Figure 5.2: Processing Time Graph.

Figure 5.2 displays an improved clustering in terms of the complexity of data-processing time; this improvement leads to better results. As the amount of data increases, the AHC takes less time to process the data in the WAN compared to the Optimized K-means.

5.4 Performance Testing

Figure 5.3 and Figure 5.4 show clustering and the SDN-enabled switch placement of both the Optimized K-means and the proposed AHC method, respectively. Each figure consists of 255 SANREN legacy nodes and the asterisks indicate the optimal position to place the SDN nodes.

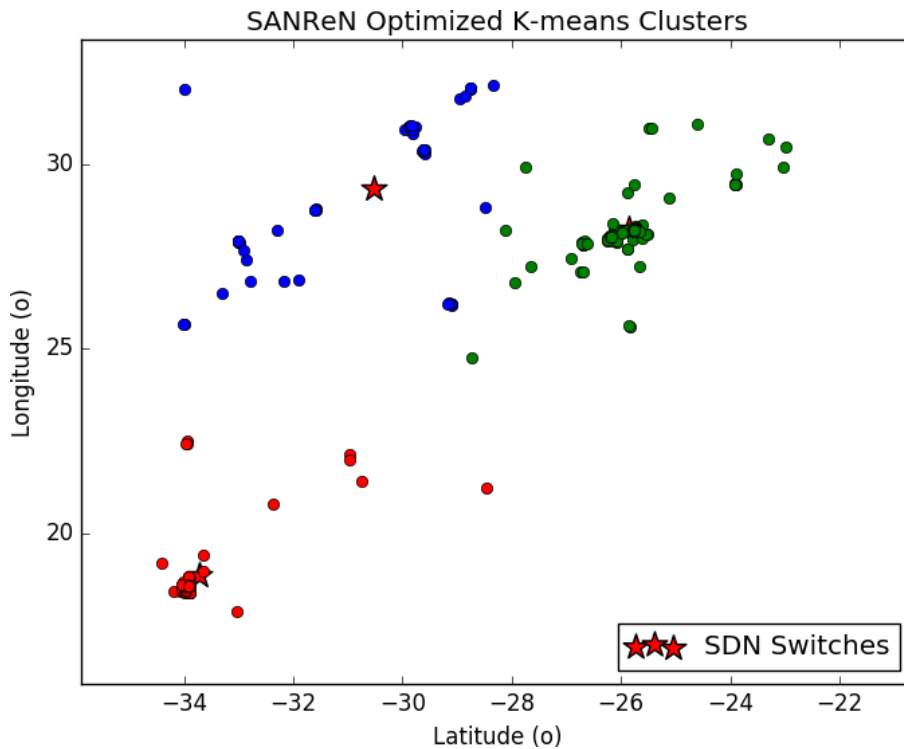


Figure 5.3: K-means Cluster.

Figure 5.3 illustrates the three clustered and ten SDN nodes network partition that is created by the proposed method. Having partitions and switches optimally placed allows for easy management and scaling-up of the network with minimal load and better traffic engineering. One disadvantage of K-means is generalization when clusters are of different densities and sizes, which is also visible on the diagram where we only see just three clusters. The technique does not recognize the density and complexity of the data provided. This can result in controllers connecting to the furthest switches instead of the closest switch for better communication. The cluster centroids can also be distorted by outliers, or the outliers can create their own cluster, therefore you would have to remove outliers before clustering, which means not the whole network dataset is considered and the outliers may be crucial to the whole network, for example, the bottom-sea and coastline nodes in the SANReN network which links to Europe and the United Kingdom. Lastly, the curse of dimensionality means that, with the increase in the number of dimensions, the similarity measure based on the distance between the nodes converges to a constant value. Therefore, this requires additional techniques such as Principal Component Analysis to set the data to linearly uncorrelated components and spectral clustering. The problems presented above clearly show how cumbersome and time consuming using the K-means technique is for real-life applicability in networks as vital as the

SANReN and commercial WANs.

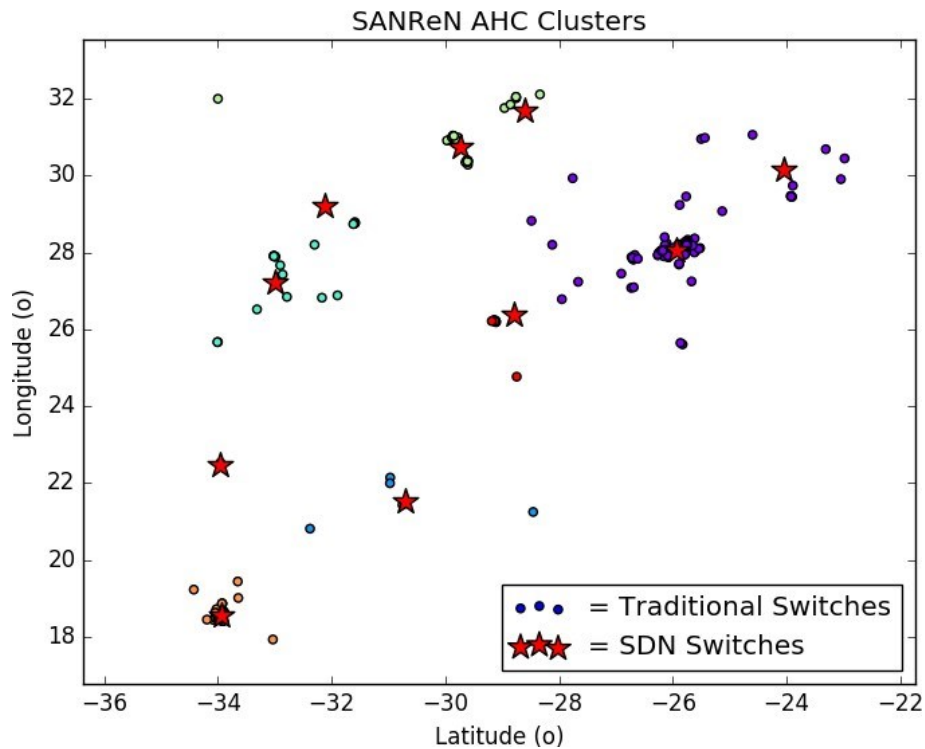


Figure 5.4: AHC Cluster.

Figure 5.4 illustrates that AHC clustering has a more partitioned network dataset compared to that of the Optimized K-means. AHC merges the neighbouring pair of clusters by satisfying similarity criteria. Until all the data is in one cluster, smaller clusters are created, which are useful in network discovery. The DI integrated with the AHC serves as a measure to find the right number of clusters on the network. The SDN-enabled switches are placed much closer to the edge of the traditional network and, most importantly, they are distributed across the WAN, thus resulting in improved delay and less loss in network performance. This allows each smaller cluster to have switches deployed efficiently throughout the centres of subnetworks of the WAN, which limits the latency between the connected devices. As shown in Figures 5.6 and 5.7, the smaller the distance between the nodes on the network, the more the throughput and the less the delay in the transmission of packets on the network.

5.5 Simulation Experiment

In order to test the validity of the algorithm results, a simulation experiment was conducted. A controller evaluation was first done by using SDN controllers; namely Ryu, Floodlight, OpenDaylight and ONOS. Figure 5.5 shows the evaluation of the controllers by using a uniform test on a hybrid network to see which controller performed better on a hybrid network. That controller could then be used to carry out the comparison of the AHC and Optimized K-means clustering algorithms. The setup included a single controller connected to one to four OpenFlow switches and 16 traditional switches. We observed the time taken to send a signal and acknowledgement signal to the source node.

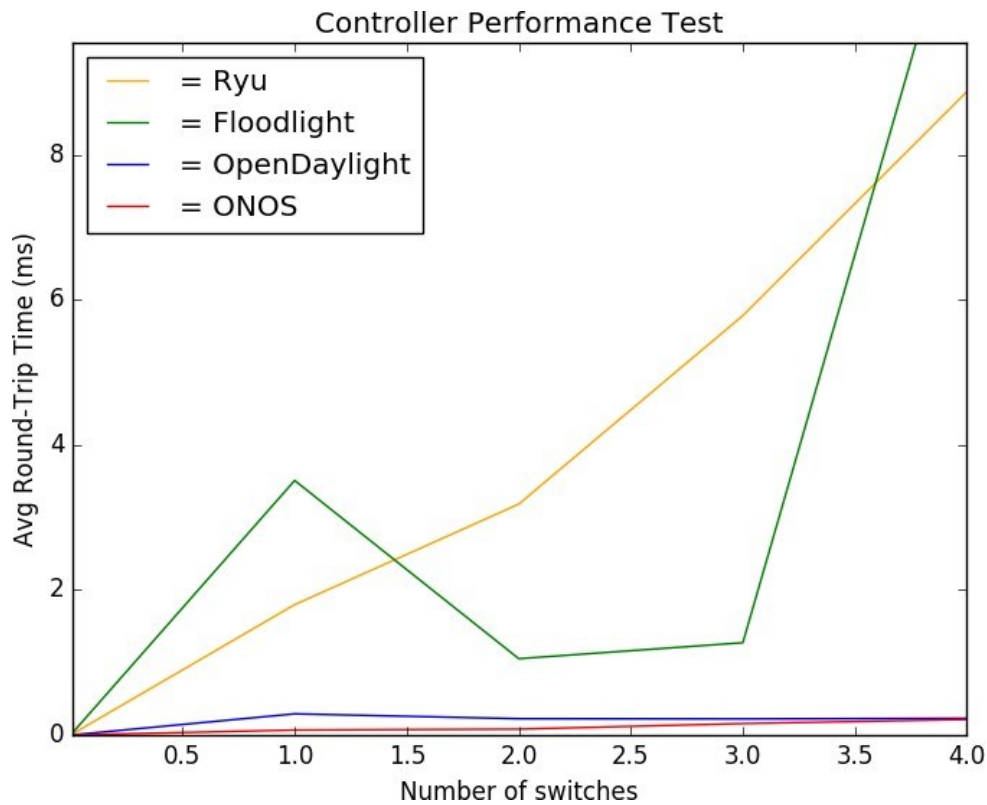


Figure 5.5: Controller performance test.

Figure 5.5 shows that ONOS has the least RTT compared to other controllers, meaning it has the least latency. Therefore it is chosen for the method test and moreover, literature by [Mamushiane, Lysko and Dlamini \(2018b\)](#) illustrates that a distributed controller is suitable for WAN to facilitate network partitioning. [Risdiyanto et al. \(2017\)](#) suggests that through the analysis of the performance results with NREN operators, an ONOS controller taking advantage of a BGP-based SDN control-plane for hybrid SDN data-plane requires less configuration and minimally affects the performance of the underlying infrastructure.

Whenever a straightforward mathematical expression can be used to establish the range to the coordinates among two points (x_1, y_1) and (x_2, y_2) , the average distance between the nodes in the optimized k-means shown in Figure 5.3 is typically 100Kms from the SDN switch to the furthest node on the WAN, and for the AHC in Figure 5.4 the average distance is typically 50Kms.

For this evaluation, the research considers the delay in transmission, the time needed to transmit a message through one point to another (Erlenback et al., 2013). Delay can be measured using equation in Section 4.3. As illustrated in Figure 5.6, one hundred pings were sent in each test and no packets were dropped. Ping test 1 represents a ping from the first to the sixteenth host, Test 2 represents the first ping to the thirty-second host.

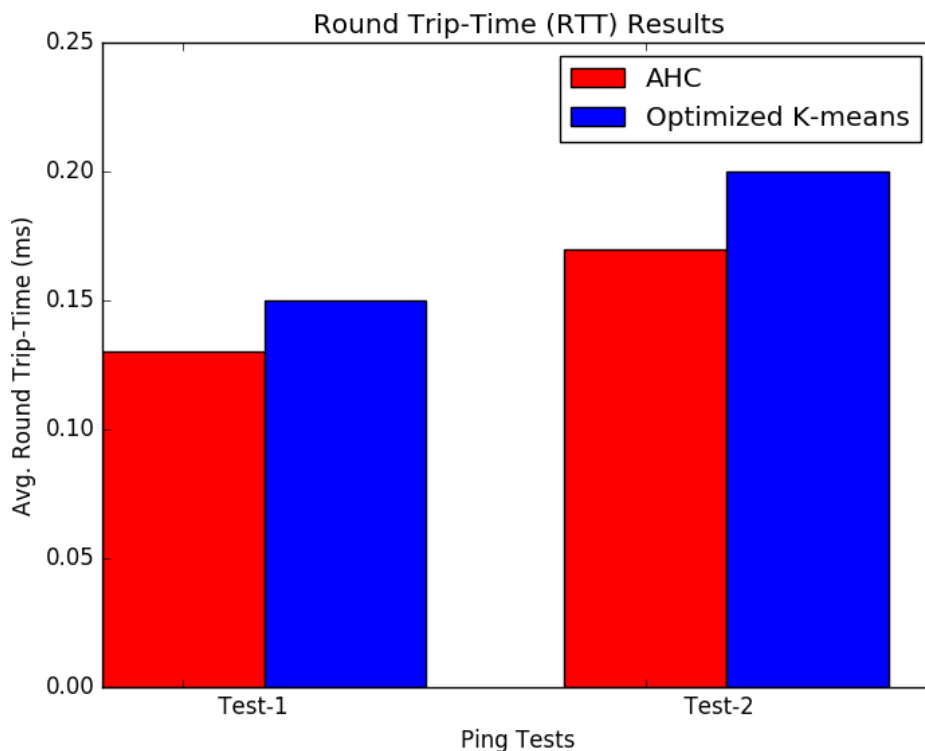


Figure 5.6: Round trip time performance results.

The average RTT is minimal on the AHC setup when SDN-enabled nodes are on the boundaries network, in contrast to the Optimized K-means scenarios in which the SDN appliances are far from conventional network boundary. In Figure 5.7, the performance test ran the throughput test on a setup similar to that of Figure 5.6.

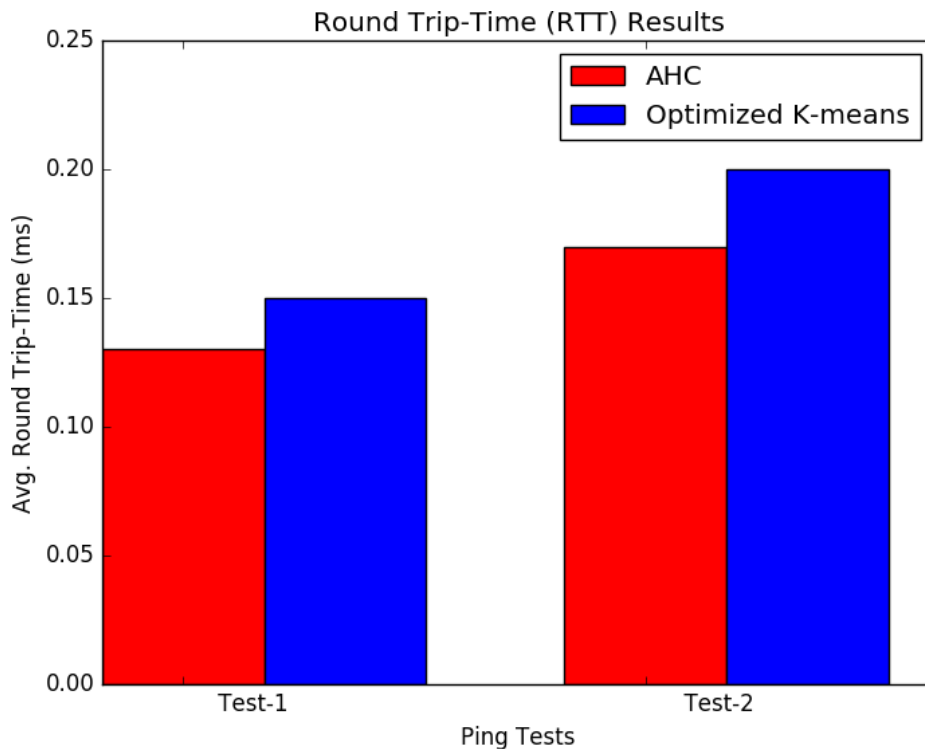


Figure 5.7: Throughput performance results.

Figure 5.7 shows that the K-means setup yields the least amount of throughput compared to the AHC, which is affected by the delay caused by the distance between SDN-enabled and traditional switches on the network.

In this section, we evaluated SDN controllers to find a suitable controller for hybrid network setup. The ONOS controller performed better for the studied use-case. The proposed method and Optimized K-means performance were evaluated in terms of their latency and throughput. The proposed algorithm performed better than the standard Optimized K-means.

5.6 Chapter Summary

The placement of SDN-enabled switches ensures effective traffic engineering which results in improved network operation. Compared to the Optimized K-means that requires the iterative insertion of the number of clusters, the algorithm automatically detects and places within each iteration multiple switches instead of a single one. In the reliability test, we observed a positive change in the latency of the proposed method compared to the Optimized K-means. The average latency was reduced to 15% compared to that of the Optimized K-means. The proposed method was also successful with a throughput of 13%.

In this chapter, the study addressed the third objective by proposing an AHC method integrated with a cluster validity measure to determine the clusters on the network to address the switch placement problem and eliminate the inconsistency related to other placement techniques. The study illustrated the effectiveness of the proposed method through simulation results which showed that the AHC effectively partitioned the network into multiple subnetworks and placed the SDN-enabled switches optimally. The AHC performed better than the Optimized K-means in both throughput and latency tests. The contribution of this work is an extension of the work in Chapter 4, by assessing an optimal algorithm to place SDN switches on a hybrid network in an emerging-market space. With switches placed optimally comes the issue of traffic engineering, where network load should be balanced through WAN links without the use of intricate routing protocols. Network load balancing becomes a vital feature when implementing networks that are centred on facilitated management and decoupled control. Chapter 6 develops and studies multipath load balancing in a hybrid SDN data plane layer to assign and distribute network traffic among accessible network properties to increase network availability, to reduce delay and bandwidth misuse.

Chapter 6

Multi-Path Load Balancing

6.1 Preamble

SDN has the capability to solve numerous telecommunications problems, but still poses many economic and business issues such as budgetary restrictions, service providers' social acceptance and support. Hybrid SDN implementation is feasible as it provides the strengths of SDN and conventional networking capabilities, along with ability to adapt to budgetary constraints, dependency on time-tested conventional methods, core network fully programmable and a gradual shift to SDN for an enterprise. As telecommunications technology becomes a necessity for the day-to-day business operations, traffic volumes have to be optimized to obtain better system performance, decrease delays and reduce the amount of time required to receive message replies. In Chapter 4, the study examined the current conventional WAN system approach and addressed reliability and stability in the case of network disruptions. And in Chapter 5 the research further studied the switching problem by identifying an efficient solution for placing SDN switches to maximize throughput and cost of maintaining SDN implementation in WAN in developing economies. The research explores and examines multipath load balancing on a combined data plane with traditional and SDN deployment

Load balancing is a method of traffic management by assigning and distributing network traffic empirically among reachable network properties to increase network accessibility and to reduce delay and bandwidth exploitation. Load balancers in SDN are implemented on the SDN controller to control traffic efficiently. The load balancer efficiency is evaluated with the use of the OpenDayLight controller with four separate hybrid SDN configurations, namely fat tree, ring mesh and torus configuration ([Badotra and Singh, 2017](#)).

This chapter's major contribution is data plane latency and connection burden management load monitoring and implementation review (before or after network failure) in hybrid SDN setting.

6.2 Proposed Solution Overview

This chapter develops and tests how well a multipath load balancer determines possible efficient flows for such a unique packet stream and equally divides the traffic volume among various efficient flows. Load balancing through hybrid SDN will introduce a workaround for rerouting network traffic and service delay on how well a multipath load balancer determines possible efficient flows for such a unique packet stream and equally divides the traffic volume among various efficient flows. Load balancing through hybrid SDN will introduce a workaround for rerouting network traffic and service delay. Depending on the expenses for each alternate connection, the effective route is chosen and statistical flows are converted to the effective route. Each minute, the algorithm changes, rendering it adaptive.

For each networking case, the load-balancing algorithm initially determines almost all of the flow routes from the very first client, by calculations the algorithm measures the expense of the route for all routes. Costs for the direction are measured as:

$$Cost = Tx + Rx \tag{6.1}$$

Where Tx is the number of transmitted packets and Rx is the number of received packets (Nkosi et al., 2018). The route with the cheapest possible cost is chosen as the shorter route, and moved to the routing table for traffic management. Load balancing is performed till the expense of the route is equivalent for all routes. The assessment check carried out in this load balancing analysis has been reran at least ten times.

The SDN controller to use for this analysis was the easily accessible-source controller OpenDayLight (ODL), because it is famous with its flexible, scalable, easily deployable and model-driven interface (Kim et al., 2017a). ODL comprises of several REST APIs for the north-bound framework, which incorporates south-bound OpenFlow-based rules. The ODL includes a configurable application system for operating network systems. The southbound framework incorporates support for various protocols including OpenFlow1.0, OpenFlow 1.3, BGP-LS etc. The components

can always be connected in Service Abstraction Layer (SAL) in actual-time. The SAL links South-bound to North-bound devices. The SAL determines the effective method for accessing requests to be handled and basic protocols used by both controller and connected switches. The controller even showed consistent performance in the throughput test via the MAC learning capability in Chapter 4.

The four environmental setups, each included ten devices, with six being OpenFlow and four conventional devices linked to eight hosts. The configurations were limited to nine considering the nature of the Torus network, with six being OpenFlow and three conventional linked to six hosts. Thus every conventional device had two hosts linked. The efficiency of the experimental load-balancing approach on hybrid networks was evaluated through the use of the above topologies. The research was using an SDN (ODL) controller, a standard unified controller (Kim et al., 2017b). The goal was to conduct an evaluation of ODL's output in hybrid networks using a multi-path load balancer (Zakia and Yedder, 2017).

6.2.1 Scenario 1: Fat Tree Hybrid SDN

The initial environmental setup in Figure 6.1 The Fat Tree hybrid configuration consists of ten devices (six OpenFlow and four conventional) and two host on every OpenFlow device. The first and furthest hosts were selected as both the set of sources and destinations. As illustrated in Figure 6.2 and Figure 6.3, Before Load Balancing (BLB) and After Load Balancing (ALB) results are illustrated below.

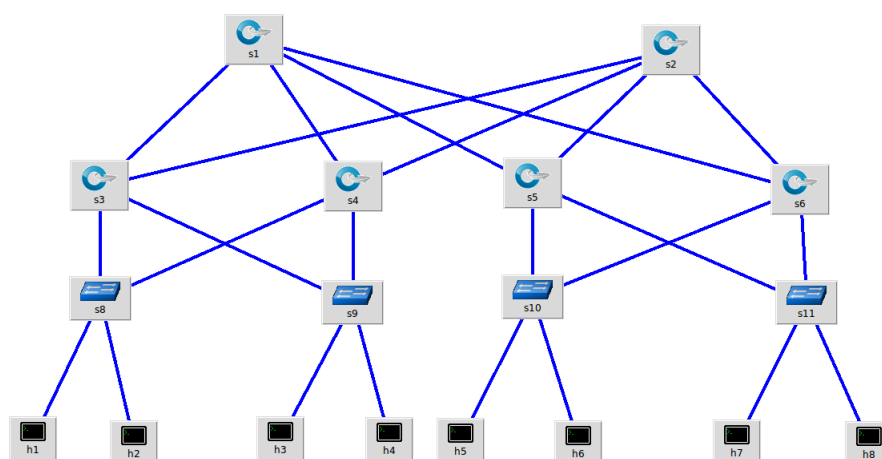


Figure 6.1: Fat Tree topology.

First, the Mininet ping output of the tests comprised of three research simulations are shown in Figure 6.2, Whereas Test 1 is H1 in switch s8 ping H4 in switch s9, Test 2 is H1 ping H6 and Test 3 is H1 ping H8. This ping analysis method is identical throughout the second and third cases, since the amount of end devices has not been altered.

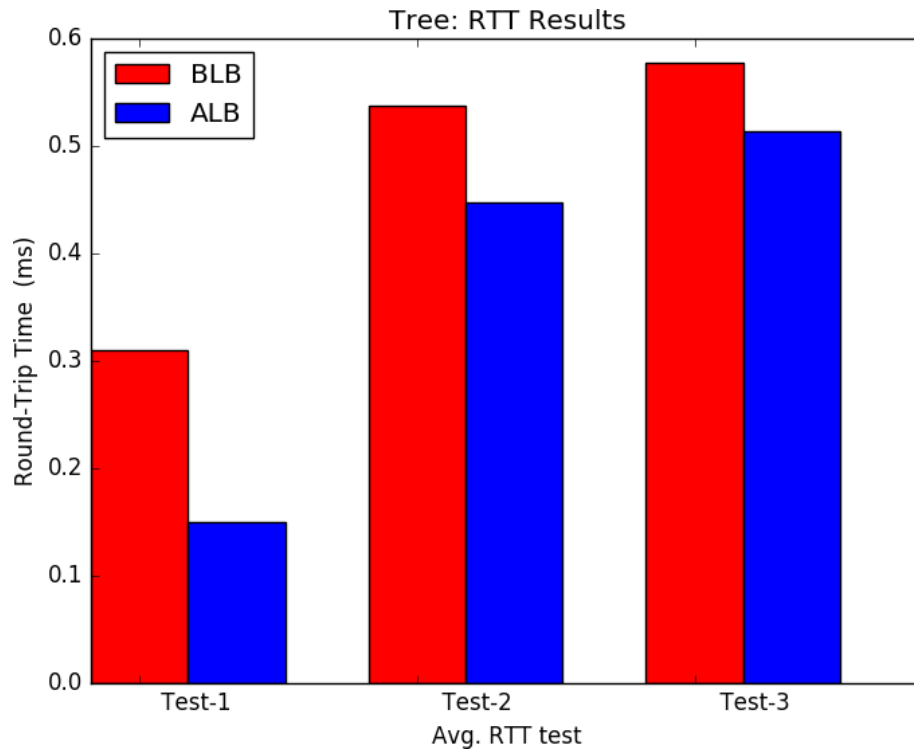


Figure 6.2: Fat Tree Round Trip Time.

In Figure 6.2, The approach suggested has been used, and thus the best and cheapest route selected was **11:: 5:: 1:: 3:: 8**, with bold numbers illustrating conventional network configuration switches. The optimal route for the source-destination set was **11:: 5:: 1:: 3:: 8**, the three ping evaluations were conducted to assess delay as the mean round-trip time (RTT) in ms on the BLB and ALB flows. Each experiment generated improved performance in RTT, with a reduction in lag following the load balancer.

Figure 6.3 shows Iperf evaluation with H1 as user and H8 as server, such an experiment was carried out to evaluate the use of bandwidth BLB and ALB.

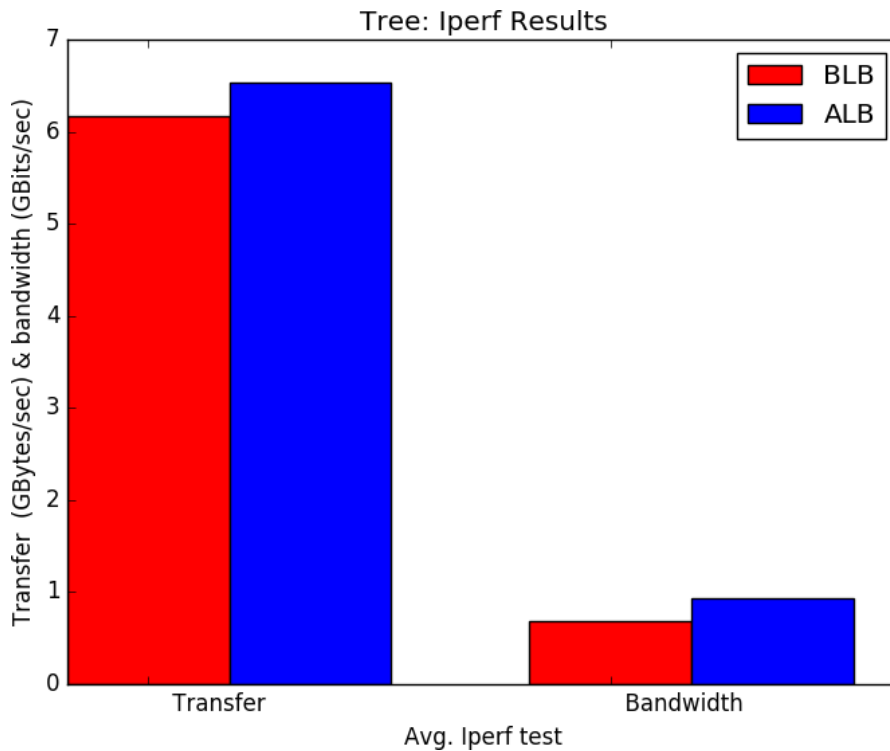


Figure 6.3: Fat Tree Round Iperf.

Figure 6.3 displays Iperf findings by a change in load balancing transmission of data. An improvement however is much as anticipated. This would be part of just the optimal route throughout the course of source-destination **11:: 5:: 1:: 3:: 8** which was congested earlier.

6.2.2 Scenario 2: Ring Hybrid SDN

The second environmental setup in Figure 6.4 the ring hybrid SDN configuration generates a single continual route on each device connecting to two additional devices. The configuration has 10 combinations for each SDN switch (six OpenFlow and four conventional devices), and two users. The load balancer picked the first one in s4, and the last user in s5 as that of the source and destination combination.

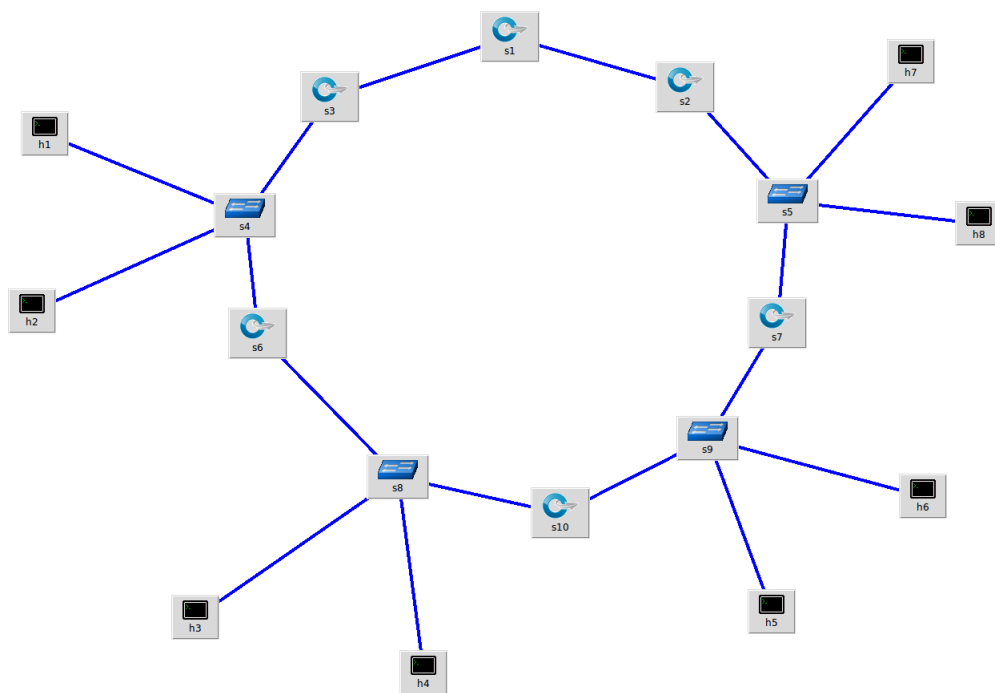


Figure 6.4: Ring topology.

The ping analysis was carried and then the outcomes of 3 experiments were identical to the findings of the first setup. Figure 6.5 and Figure 6.6 show the RTT and Iperf results of the ring topology.

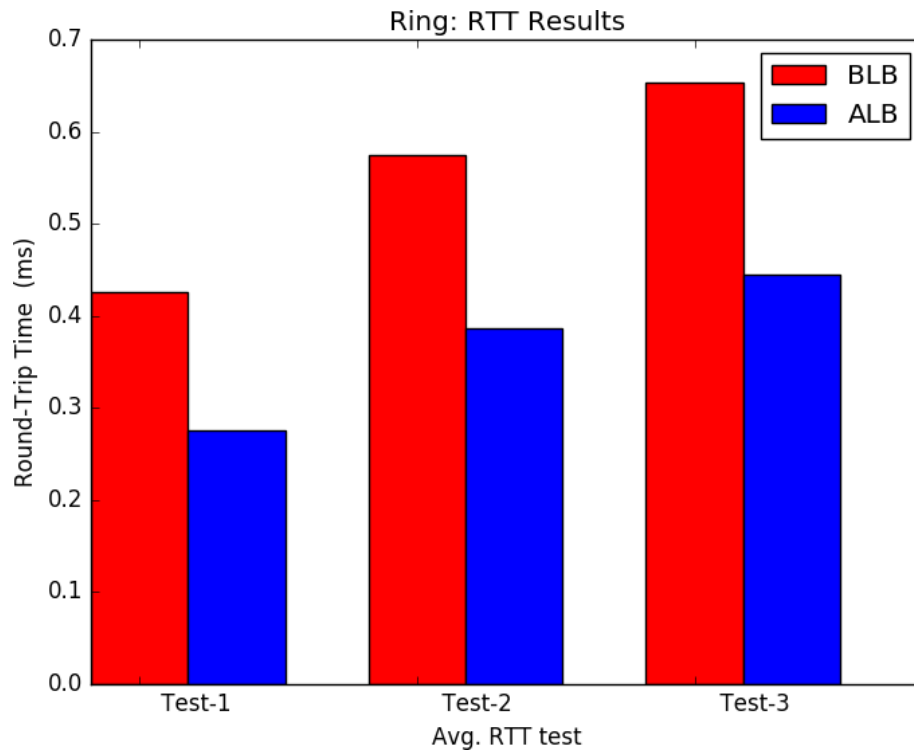


Figure 6.5: Ring topology Round Trip Time.

In the second setup, the route of the selected source-destination set was **9:: 7:: 5:: 2:: 1**. the output was calculated BLB, and the output ALB had also been tested after creating traffic in the alternative route. Using the load balancer in a ring hybrid setup, the latency also dropped significantly

Figure [6.6](#) serves to illustrate ring configuration Iperf source and destination set effects throughout the BLB and ALB user and server comparison for bandwidth consumption estimation.

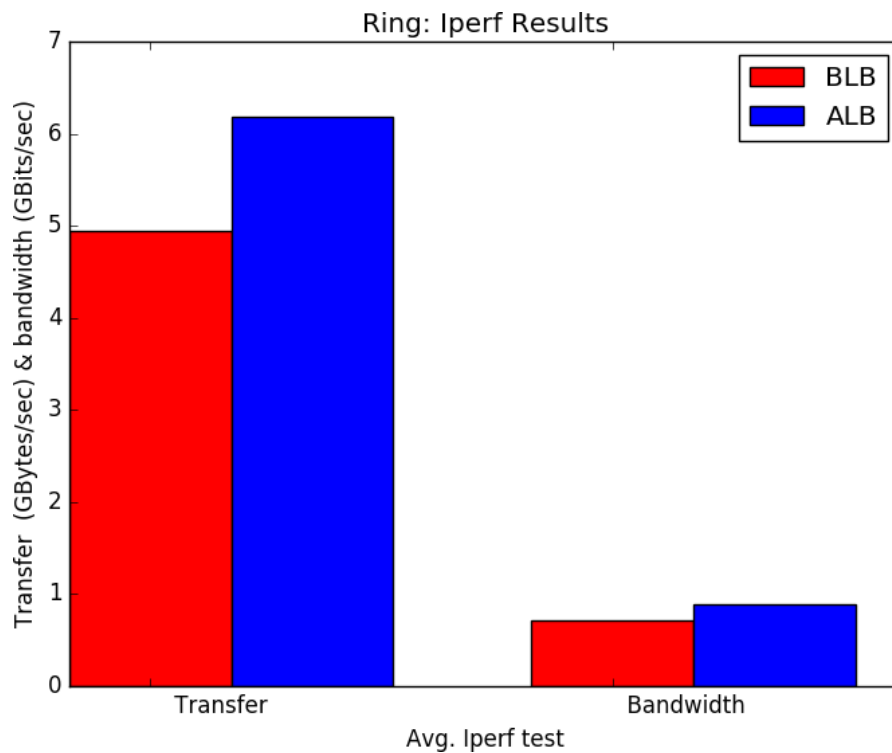


Figure 6.6: Ring topology Iperf.

The Iperf analysis in Figure 6.6 shows an improvement in ALB transmission of data by about 1.25 GB. That's also because the source-destination set is used to record the Iperf performance.

6.2.3 Scenario 3: Mesh Hybrid SDN

The third environmental setup shown in Figure 6.7 is a mesh hybrid configuration in which devices are linked, where not only each node reroute its own traffic, it also transmits information from several other nodes, including ten switches (six OpenFlows and four traditional ones).

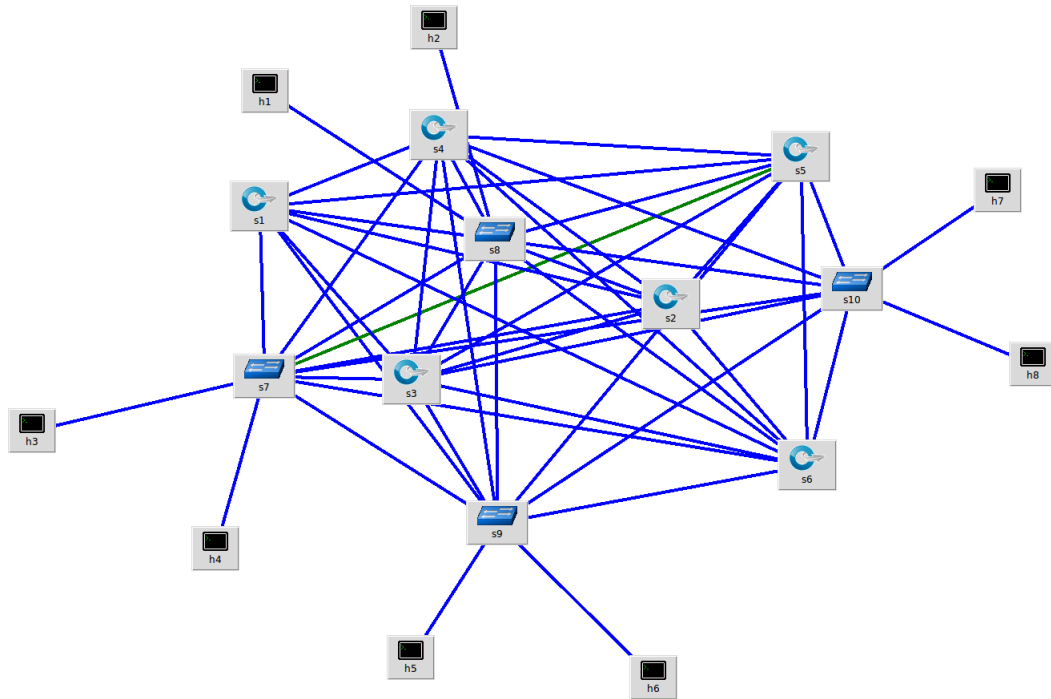


Figure 6.7: Mesh topology.

The initial and last hosts have also been identified as the set of origins and destinations. Mininet ping check and the outcomes of the three experiments were identical with those of the cases of the first and second experiment. Figure [6.8](#) and Figure [6.9](#) illustrates findings of an assessment of the mesh topology.

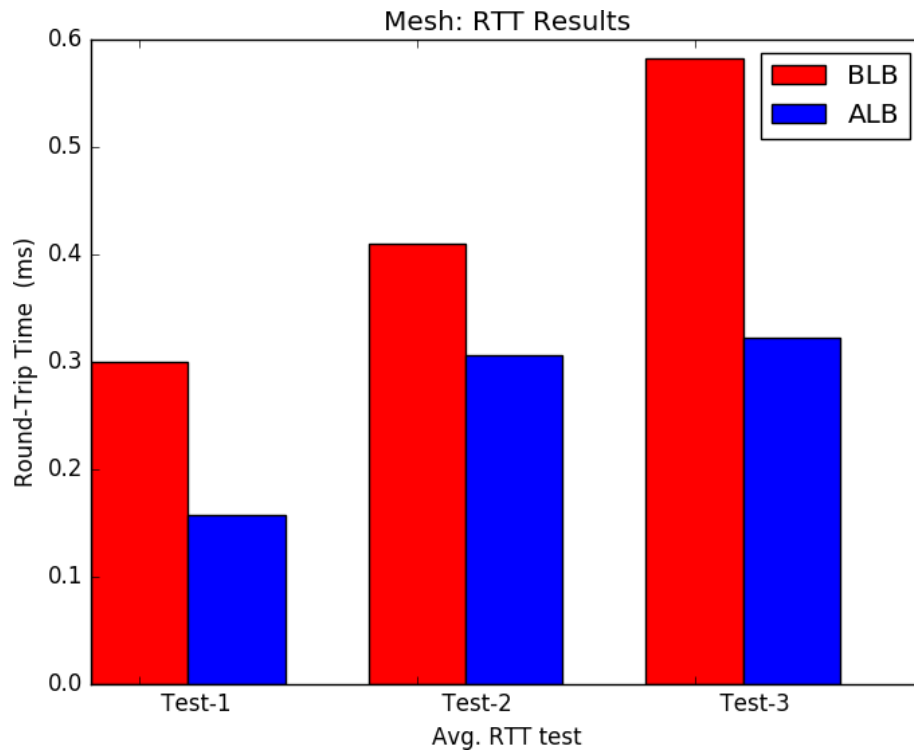


Figure 6.8: Mesh topology Round Trip Time.

The source destination of mesh topology shows a straightforward path of 9::7 because of its node connectivity mechanism, in which the device takes the shortest alternative route directly towards its endpoint. This might also be the explanation why there is a reduction in RTT, particularly from the farthest device H8 ping, since it was also defined as the least cost route. The success of the third evaluation which is better than other experiments could be clarified by means of intricately interconnected relations which can contribute to the shortest route from the source device to its target. This is among the results of the shorter route estimation, which prevents repetitive network hops.

Figure 6.9 demonstrates network configuration of H1 to H8, host and server evaluation outcomes of BLB and ALB bandwidth usage measurement.

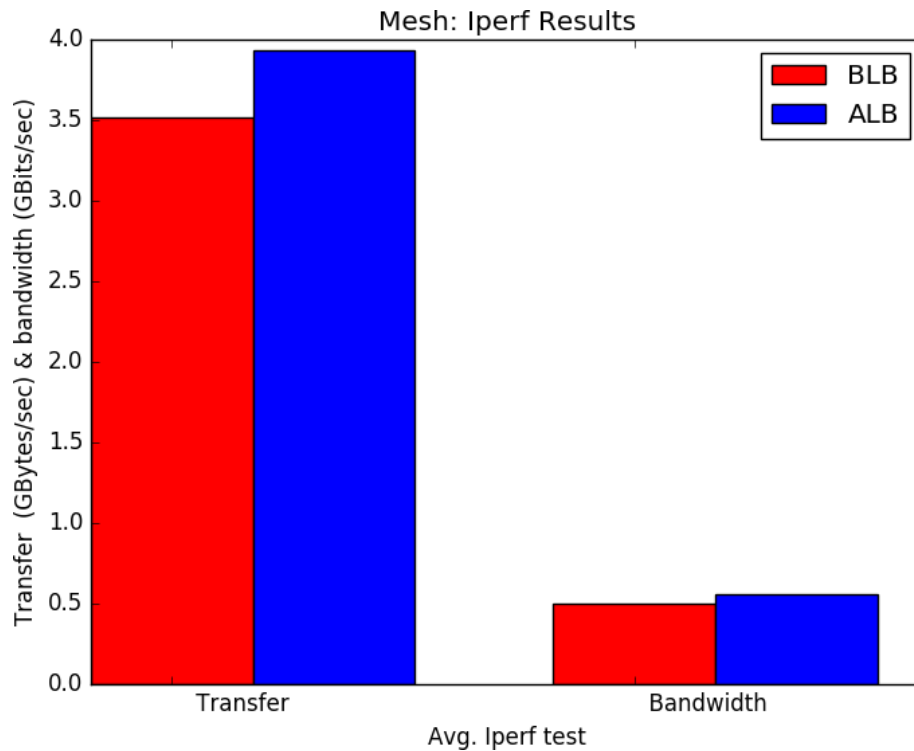


Figure 6.9: Mesh topology Iperf.

The transfer rate and bandwidth utilization in Figure 6.9, because of the similar explanation of device interconnectedness, hardly any substantial increase was shown, however the load balancer helped identify the shortest and less expensive route. This led to a rise in transmission of data of around 0.4 GB / s ALB. The outcomes as well demonstrates a slightly better use of bandwidth forward to load balancing.

6.2.4 Scenario 4: Torus Hybrid SDN

The last environmental setup in Figure 6.10 is the 2D Torus hybrid SDN configuration (Askar, 2016), that is centred on a 2D-mesh with additional links. Torus differs from other configurations because it comprises of nine devices (six OpenFlows and three traditional switches).

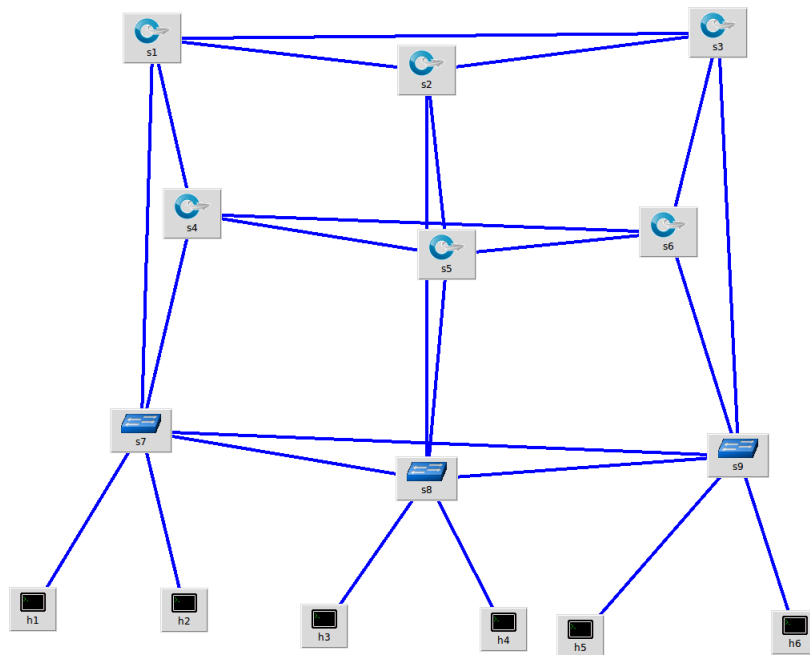


Figure 6.10: Torus topology.

The initial and the last hosts have also been identified as source and destination set for this configuration which is H6. Mininet ping experiment and the findings of both the three ring topology analyses RTT and Iperf are illustrated in Figures [6.11](#) and [6.12](#).

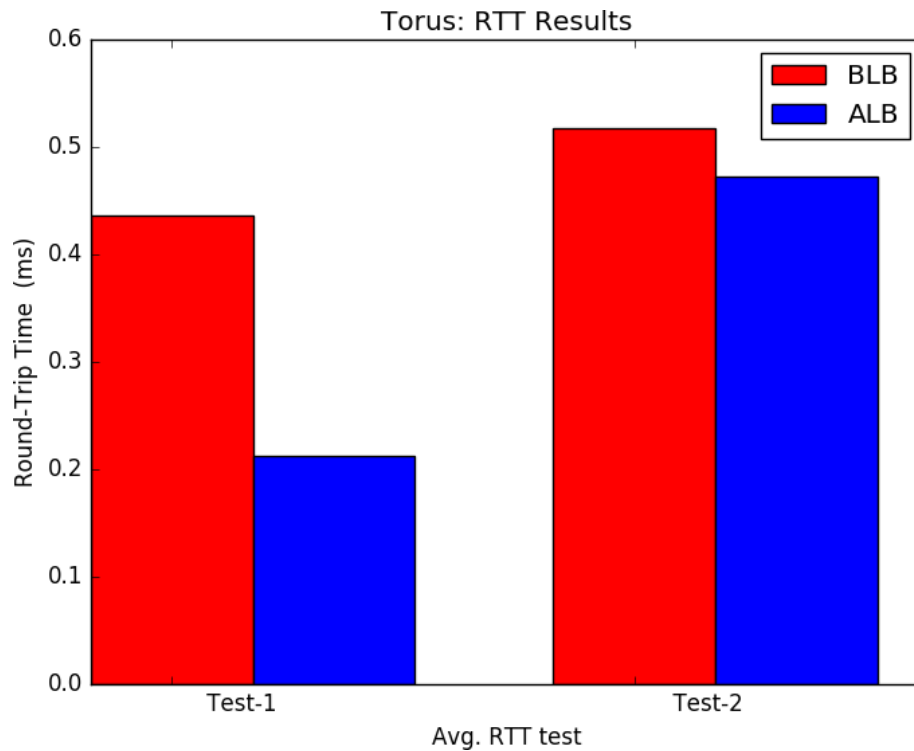


Figure 6.11: Torus topology Round Trip Time.

Figure 6.11 shows the Torus hybrid configuration output in RTT, the source-destination route also is clear as the mesh route. 2:: 1. Throughout this analysis the two ping assessments unlike most other configurations; H1 ping H4 as Test 1 and H1 ping H6 as Test 2. The load balancer demonstrates a considerable improvement in delay, particularly during the first assessment. This seems to be related to the direction with the lowest mean expense of 40 per cent in comparison to the final evaluation.

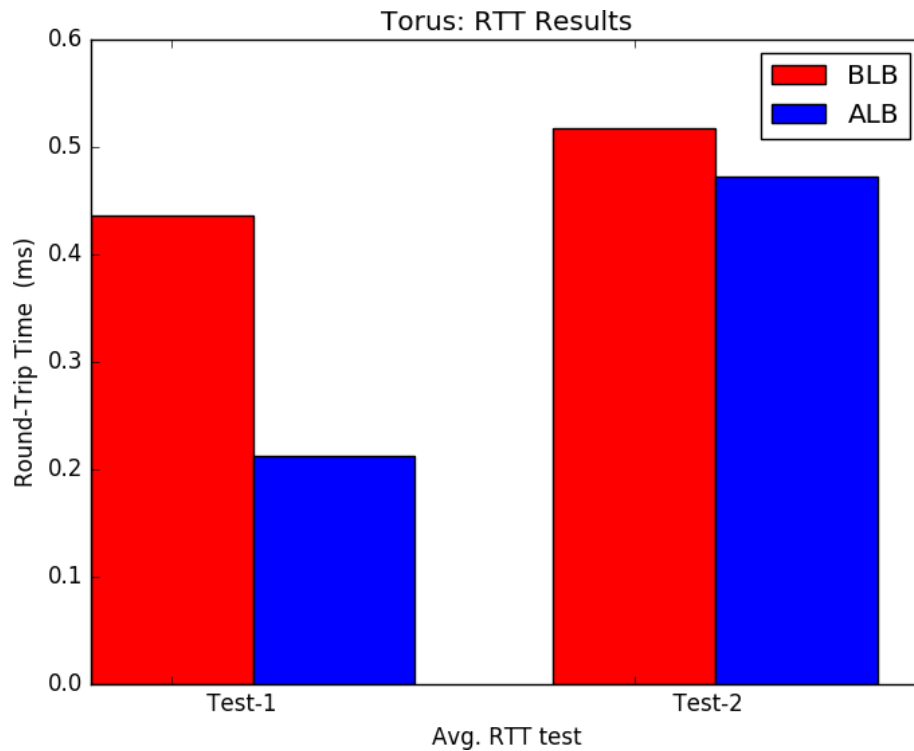


Figure 6.12: Torus topology Iperf.

In Figure 6.12, the findings of the Iperf check again revealed no change however the load balancer increased throughput. The study did not demonstrate any variation as there was no higher proportion of devices connected in the configuration to display several hops. However towards the end, the load balancer accomplished what had been anticipated, with an improvement of 0.3 GB / s.

According to the above experimental results, this is clear that perhaps the load-balancing approach improves system performance through minimizing latency and that the usable path bandwidth by 0.2 Gbps and decreasing packet loss. This shows that now the presented load-balancing method for multipaths will enhance quality of service and allow effective usage of the lower level of infrastructure.

6.3 Chapter Summary

This chapter evaluated the delay, transfer speed and the bandwidth utilization across four different topologies tested in hybrid networks. The applied algorithm used the shortest path approach for computing the path to which the traffic can be engineered. When tested against an ODL controller load balancer, the proposed algorithm performed much better than the controller load balancer. The performance was tested in

simulations of the topologies, and the tree topology showed an improved average RTT of 0.4 ms in all the tests and showed improvement of 0.2 to 0.3. GB/s in transfer speed and 0.2 Gbps in bandwidth utilization.

The ring topology had a simpler setup to that of other topologies which resulted in an average of 0.6 ms RTT, the transfer speed was improved by 1.2 GB/s and the bandwidth by 0.2 Gbps which represented impressive results for the load balancer. The mesh topology was more complex than the other topologies under evaluation, the average RTT test showed an improvement of 0.5 ms, with a transfer speed of 0.9 GB/s and a 0.1 Gbps of bandwidth utilization. Lastly, the torus topology, as the name suggests, takes a chainlike structure that can interconnect nodes and can connect the source node to the destination node without going through other nodes, which explains Test 1's improvement in RTT, with an average of 0.6 ms. The transfer speed of Torus was not too much because the test targeted the farthest node, which resulted in a close call of 0.1 GB/s improvement. Bandwidth utilization was not very great in terms of efficiency at 0.2 GB/s.

To the best of our knowledge, there is a shortage, if not an absence, of studies that have proposed such a technique and applied the technique and evaluated four different kinds of topologies. Therefore, this work has contributed to the hybrid SDN space by developing and evaluating a solution for load balancing in hybrid networks in emerging markets by using four types of network topology. This study expands on the studies in Chapters 4 and 5 on hybrid networks aimed at finding solutions for the optimal deployment of SDN in emerging markets. The next chapter looks at the financial challenges and provides a project plan together with the quantifiable analysis of the estimated cost of migrating to SDN through hybrid SDN and the time it would take an organization to recover the estimated CAPEX involved in the migration.

Chapter 7

Hybrid SDN as an Enabler for African SD-WANs

7.1 Preamble

The research and education advancements in Africa have resulted in National Research and Education Networks (NRENs) having to spend millions on upgrading facilities and improving network availability for all. Unlike Internet service providers (ISPs), NRENs do not support commercial infrastructure sharing, therefore a lack of telecom infrastructure and no government grants to be in operation are some of the difficulties these networks face. Software-Defined Networking (SDN) promises central management of the network architecture, which can compliantly allocate network resources to existing network infrastructure through hybrid SDN capabilities. While Europe and North America are technologically advancing in academia, Africa remains comparatively behind the National Research and Education (NREN) pack owing to insufficient funds for expanding and purchasing advanced telecommunication infrastructure. The purpose of these networks is to make available access for academic and scientific needs and to facilitate partnerships in education, knowledge acquisition, and research. These partnerships necessitate focused security and sometimes services that are not commonly accessible or are too luxurious over an ISP (Foley, 2016). Functioning expenditure on NRENs are met through the membership fees and provision charges. However, the government funds key expenses such as primary setups and subsequent improvements.

Connectivity is expensive. Research is done by Research ICT Africa (RIA) suggests that the price tag of the internet is creating obstacles to connecting the low-income population to the net.

Because 50% of South Africans fall in the low-income group, 50% of the population is without access to the internet ([Gillwald, 2010](#)). With the focus on Africa, it is evident from ([Foley, 2016](#)) that Bandwidth prices have fallen from “US\$4, 000–6, 000 per megabit per second (Mbps) per month to below US\$100 per Mbps per month” as some African countries have taken the initiative to migrate from satellite to fibre optics. This shows a positive change in technological advancements at reduced operational costs.

Having analysed placement and networking engineering in Sections 4 to 6, it is important to know the financial implications of the migration mechanism. This section aims to investigate the quantitative influence of hybrid SDN as a migration strategy to SDN in NRENs with a focus on Africa as an emerging market. The impact of SDN on Capital Expenditure/Operational Expenditure (CAPEX/OPEX) is quantified for the government business case, as it is considered to be the primary source of funding because it is responsible for capital expenses for primary setups and future upgrades.

7.2 Proposed Solution

This chapter is an extension of the work by ([Mamushiane, Lysko and Dlamini, 2018a](#)) using a financial model which considers emerging markets through capital budgeting. The study assesses the financial implications associated with migration from traditional networks to SDN networks through hybrid SDN in the African NREN space. However, this study explores a more refined model that considers the time value of money, various financial factors and NRENs as a non-commercial enterprise.

The first scenario is a traditional network setup, before the deployment of SDN devices where the network is operating entirely on legacy equipment. The second scenario is a hybrid SDN-enabled network setup, where the SDN-enabled devices are deployed in the same setup environment with traditional network devices. In the second scenario, the network infrastructure consists of SDN-enabled devices coexisting with legacy devices.

There are various techniques of capital budgeting, of which one of the traditional methods is the payback period method which characterises a non-complex approximation of the time taken by a company to receive the asset cost repaid ([Mamushiane, Lysko and Dlamini, 2018a](#)) This is calculated as;

$$\text{Payback period} = \frac{\text{Initial Investment}}{\text{Periodic Cash Flow}} \quad (7.1)$$

The lengthier the equalling period, the greater the risk associated with the project. However, this technique has a major flaw because it overlooks the time value of money. It also overlooks the cash flows subsequent to the payback period (Beckett-Camarata, 2009).

However, the discounted payback period technique is also calculated as;

$$\text{Discounted Payback period} = \frac{\text{Actual Cash Flow}}{\text{Previous Value Factor}} \quad (7.2)$$

The current value factor is defined as $1 + in$, where i is the discount rate and n is the period. This technique considers the time value of money, which discounts any upcoming cash flows to current value and associates the investments on a payback basis (Ardalan, 2012). This method means a single rand currently is valued at more than one rand by the end of the year.

In the cost analysis, the study considered the following financial factors; Inflation adjustment and/or industry rate of price escalations of IT products for hardware and software support, as well as the escalation clauses on the rental agreement., Rent normally escalates by 10% per annum. The National Energy Regulator of South Africa (NERSA) approved rates guaranteed to Eskom (South African electricity public utility) at a mean price rise of 13,87 percent to be introduced for Eskom's individual customers on 1 April 2019 and an mean price hike of 15,63 percent for municipalities on 1 July 2019 (Karakus and Duresi, 2019). The cost-of-living adjustment (COLA) uses industry norms at a projected 5.2% consumer price index (Ardalan, 2012), which will provide the average salary increase annually.

Assumptions

The experimentation is tailored for African NRENs.

- The infrastructure analysed is assumed to be that of a new installation of SDN-enabled devices.
- The assumption is based on a non-standard procedure or technique for deploying hybrid SDN.
- The nature of hybrid networks should be taken into account for a fair comparison of existing networks with the SDN-enabled switches as the assets.

7.3 Quantitative Evaluation

In this chapter, quantified evaluations are done for the migration to SDN in African NRENs in order to assess the financial planning required and the time it would take for such technology to be profitable in the African NRENs space. As mentioned in the previous section, capital budgeting is used for the analysis.

Initially, a project costing estimate is done to predict the development cost of a hybrid SDN approach to full SDN deployment in African NRENs. Figure 7.1 shows the three-phased project’s cost estimate, with details of the duration of each phase.

	Phase I			Phase II			Phase III	
	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2
CAPEX								
Site Acquisition and Design	Red	Green	Green	Green	Green	Green	Green	Green
Router Pricing	Yellow	Red	Red	Green	Green	Green	Green	Green
Building, Rigging and Materials	Yellow	Red	Red	Red	Green	Green	Green	Green
Power Installation	Yellow	Red	Red	Red	Green	Green	Green	Green
Network Switches	Yellow	Red	Red	Red	Green	Green	Green	Green
Backhaul	Yellow	Red	Red	Red	Green	Green	Green	Green
SDN Controller Pricing	Yellow	Yellow	Red	Red	Green	Green	Green	Green
Network Testing	Yellow	Yellow	Red	Red	Green	Green	Green	Green
OPEX								
Hardware and Software Support	Yellow	Yellow	Yellow	Red	Red	Red	Red	Red
Land Rent	Yellow	Red	Red	Red	Red	Red	Red	Red
Power	Yellow	Red	Red	Red	Red	Red	Red	Red
SDN Certified engineers	Yellow	Yellow	Yellow	Yellow	Red	Red	Red	Red
Maintenance	Yellow	Yellow	Yellow	Yellow	Red	Red	Red	Red
Insurance	Yellow	Yellow	Yellow	Red	Red	Red	Red	Red
Contingency	Yellow	Yellow	Yellow	Red	Red	Red	Red	Red
Management	Yellow	Yellow	Yellow	Red	Red	Red	Red	Red

Figure 7.1: Project Plan Gantt Diagram.

In Figure 7.1, red represents the active process, yellow represents a process waiting for another process to complete and the green represents a completed process.

The analysis uses the Gantt chart alongside a spreadsheet to estimate the running cost of each process in order to conduct a CAPEX and an OPEX. The analysis illustrates that with the existing passive CAPEX with components such as building, rigging and materials, costs can be reduced on average by 10-20% of CAPEX and 25-30% of OPEX costs. A supplementary 15-20% CAPEX and 15-10% OPEX are also reduced by sharing the traditional backhaul. Lastly, sharing the core network would reduce CAPEX and OPEX by an extra 15% and 20% respectively.

Figure 7.2 illustrates the output of the discounted payback period, which discounts any upcoming cash flows to current value and associated investments, with regards to the financial factors referred to in the previous section. The graph presents the number of months to cover CAPEX and OPEX costs for each project to become profitable.

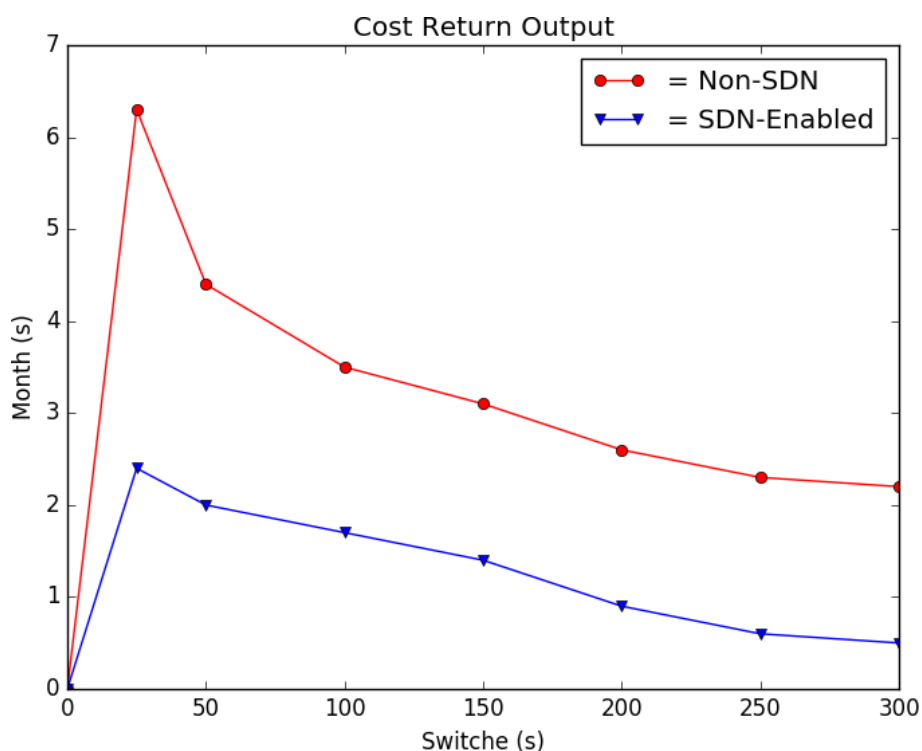


Figure 7.2: Cost Estimation Graph.

From the results in Figure 7.2, it is clear that the existing traditional network environment of conventional switches and management of infrastructure in African NRENs can lead to significantly delayed payback periods. These results may have been among the main reasons for delayed SDN rollout, owing to insufficient funds to cover new projects and distribution of capital-extensive and poor quality service in

African NRENs. Moreover, this has resulted in a tough entry barrier for new entrants who do not have the CAPEX required for network rollout.

Table 7.1 shows the effectiveness of the hybrid approach to CAPEX recovery time. The results illustrate that having a hybrid deployment yield a significant drop.

Table 7.1: Hybrid CAPEX and OPEX recovery

	Hybrid switches	Backhaul	Core	Total
CAPEX	20%	10%	15%	45%
OPEX	15%	7.5%	12.50%	35%

The hybrid approach CAPEX results are improved by an additional 20% minimum. The reduced backhaul operational expenses are due to the optimal placement of SDN-enabled switches and the use of the existing links from the legacy infrastructure while taking advantage of central management through SDN features such as protocol translation, where SDN devices can take control of the legacy packet to the SDN controller, parse packets, and inject packets from the SDN controller to the network. The hybrid direction would minimize the entry barrier and will stimulate productivity at service level instead of infrastructure level, thereby driving African NRENs to seek innovative ways to improve the quality of their service. This approach is likely to reduce subscription costs dramatically in emerging markets. Ultimately, this is likely to pave the way to bridging the digital divide in an emerging market to foster economic growth.

7.4 Recommendations

First, the government should make available operator's licences to the NREN in accordance with the acceptable use policy. Such policies will guarantee that somehow the NRENs will not interfere with the private industry in supplying the commodity connection to the citizens, this would allow NRENs to exploit technologies such as SDN-enabled infrastructure sharing in emerging markets, which has been proven to have a positive influence on organizational CAPEX and OPEX.

Despite the continuously increased competition for internet in African countries, contending on monetary value alone should not be the NREN's foremost or unique objective. Their significance for their member universities is and will still be based on their capability to deliver more and more relevant services such as SDN and the next generation of networks, 5G, with the focus on education and research, which are progressively global and collaborative. Hence, it is important that NRENs invest in skills development programmes for their engineers to be able to deliver advanced services.

Lastly, NRENs should engage more with decision-makers and exchange knowledge to raise awareness among them of the importance and nature of NRENs in order to get support from growth associates such as the AAU, the European Union (EU), and European NRENs who have more developed NREN infrastructure and have explored the migration to SDN.

7.5 Chapter Summary

The section highlights the significance of enabling hybrid SDN in WANs for African NRENs to optimize capital and operational expenditures. A financial plan has been developed for a three-phased migration, and a mathematical model to calculate the cost of certain parts of a network to estimate the quantifiable influence of the hybrid approach to software-defined wide area networks in African NRENs. The results suggest that with the existing traditional infrastructure CAPEX resources such as building, rigging and materials can reduce the cost of capital when migrating to SDN by using existing infrastructure up to an average of 10-20% and another reduction of about 25-30% in OPEX costs. The cost of a supplementary 15-20% CAPEX and 15-10% OPEX is also reduced by sharing the traditional backhaul. SDN offers profitable use of computing resources for network administrators, and facilitates resource requirement. While bringing fully programmable easiness through modifying the attributes of the whole network.

It is evident that hybrid SDN implementation can reduce the time to reach the profit margin from 4.5 months to less than 1.4 months on average. This economic shift can result in a range of captivating prospects, such as lower costs to the end-users, and enabling smaller research and education institutes to enter the NREN family. In future, the study intends to expand our analysis to cover other African use cases, also in scenarios where there already is SDN infrastructure.

Chapter 8

Conclusion and Future Work

8.1 Preamble

Software-Defined Networking (SDN) can solve the present day multidimensional networking challenges but certain implementation problems occur. With mainstream network implementations in general, a seamless transition to full-SDN rollout needs to satisfy budgetary issues, which can allow easier, cheaper and less costly failsafe into existing resources to build trust. These advantages can bring change to network advancements in the slowly transitioning emerging markets as the future of the network and linked global communication looks towards the emergence of 5G and the internet of things in the African setting and other emerging markets.

The management of WANs is characteristically costly and complex. These disadvantages require specialized and pricey high-performance routers with elevated CapEx, along with complex design involving expert maintenance activities, contributing to total OpEx and risk of failure. In comparison, WANs often dependent on primary transport network connections that are prone to error. Centrally controlled SDN and network governance will help to even more efficiently distribute network capacity, dynamically adjust to changing connectivity demands, and much more swiftly handle failures. That can support WANs by growing the required over-supply and complexities. Challenges of the duration the packet arrive at its target as well as the time required to analyse the packet header are somewhat prevalent throughout WAN, where QoS is much more favoured. Thus, it is necessary to identify the optimum switch location and the number of devices to be deployed for a specified network configuration magnitude, such that the total performance output is improved while preserving a fair load density on the network.

8.2 Contribution of Knowledge

This research provided a workaround for the current conventional WAN framework to meet the first and second goals of the project objectives in Chapter 1, by targeting at providing versatility and stability throughout the occurrence of system disruptions via a decentralized SDN system overlay and boundary SDN nodes in Chapter 4. Throughout the flexibility test in Section 4.3, ONOS beat all other controllers by providing the maximum throughput while managing the increased amount of network devices and MACs. The study revealed a meaningful change throughout the RTT in the ONOS reliability experiment in Section 4.4, when deploying three controllers instead of two, with overlay devices on the boundaries of the network. The solution was presented in Chapter 4, where the proposed solution yielded improved results in terms of flexibility and reliability. The flexibility test on controller performance was carried out to assess the optimal controller in hybrid WAN. The total RTT was lowered to very little than 0.2ms and, more significantly, this eliminates the bottleneck. In the perspective of centralized management, boundary overlay node location, and also latency, a versatility solution focused on SDN overlays is practical for further implementation throughout WANs. The studied metrics on RTT and throughput showed improvement in the reliability test equal to averages of 1.182 and 2.166 ms with two and three-controller deployment respectively and a throughput average of 20 GB/s.

To deliver convenient and effective traffic engineering, resulting in network performance improvements, the study has proposed an AHC method integrated with a cluster validity measure to determine the clusters on the network in Section 5.2.1. This technique addresses the switch placement problem and eliminates the inconsistency related to other placement techniques to address the third objective 2 in Section 1.5.2. The study further illustrated the effectiveness of the proposed method through simulation results in Section 5.3, which showed that the AHC partitioned the network effectively into multi subnetworks and optimally placed the SDN-enabled switches. In the reliability test, the results showed a positive change in the latency of the proposed method compared to Optimized K-means. The average latency was reduced to 15% compared to that of the Optimized K-means. The proposed method was also successful in the throughput (with an improvement of 13%). In the future, the study on the project planning and techno-economic analysis would form a basis for the initial stages of a migration process to a fully SDN African NREN environment. The placement of SDN-enabled switches ensures effective traffic engineering, which will result in improved network operation. Compared to the Optimized K-means that requires the iterative insertion of the number of clusters, the algorithm automatically detects and places multiple switches instead of a single one in each iteration.

To further optimize the strategy to solve the switch placement problem in emerging markets to address the fourth objective in Chapter 1, the four separate hybrid network configurations in Section 6.2 were tested utilizing a load-balancing multi-path framework on the data plane. The load balancer has been identified to significantly increase network performance across all four network configurations from Section 6.2.1 to 6.2.4. Nonetheless, it was found that perhaps the data plane configuration undergoing the study would have to constitute of a larger amount of switches in order to have more alternate connectivity routes for better performance and straightforward assessments. The tree topology yielded an improved average RTT of 0.4 ms and 0.2 to 0.3 GB/s in transfer speed and 0.2 Gbps in bandwidth utilization in Section 6.2.1. The ring topology resulted in an average of 0.6 ms RTT, the transfer speed of 1.2 GB/s and bandwidth of 0.2 Gbps in Section 6.2.2. The mesh topology resulted in an average RTT test of 0.5 ms, with a transfer speed of 0.9 GB/s and a 0.1 Gbps of bandwidth utilization in Section 6.2.3. The Torus topology in RTT yielded an average of 0.6 ms in Section 6.2.4. The transfer speed improved by 0.1 GB/s and bandwidth utilization was not very efficient at 0.2 GB/s.

Moreover, the significance was highlighted of enabling hybrid SDN in WANs for African NRENs to optimize capital and operational expenditures in Chapter 7. A financial plan was developed, for a three-phase migration, and it was used to estimate the economic influence of the hybrid approach in software-defined wide area networks in African NRENs in Section 7.3. The impact of SDN integration has been estimated in terms of the number of months required from starting a new infrastructure rollout to making it profitable in Section 7.3. It is evident that hybrid SDN implementation can reduce the time to reach the profit margin from 4.5 months to less than 1.4 months on average. This economic shift can result in a range of captivating prospects, such as lower costs to the end-users, and enabling smaller research and education institutes to enter the NREN family as outlined in Section 7.4.

8.3 Conclusion

The research study has presented a hybrid SDN migration approach to accelerate SDN deployment in emerging markets by using four phases of deployment: Placing the SDN switches on the edge of the metros on a WAN; Finding optimal locations of the placed switches by using cluster techniques. Balancing the load of the network, and conducting an economic analysis of the migration. These were achieved in response to the research questions in Chapter 1 section 1.4 which were based on the objectives set for this study in Chapter 1 section 1.5.

The study focused on these objectives to address objectives 1 and 2 in response to the first, second and third research questions. The contribution of chapter 4 was focused on finding an optimal deployment solution to the transition to SDN, where the study proposed a solution that provided flexibility and facilitated management by reducing delay and using bandwidth efficiently. Furthermore, this work also assessed open-source controllers to find a suitable controller for such scenarios and the study contributed by addressing the controller scalability problem. To address the fourth research question through the third objective, Chapter 5 assessed an optimal algorithm to place SDN switches on a hybrid network in an emerging market space. The fourth objective was used to address the fifth research question. This work has contributed to the hybrid SDN space by developing and evaluating a solution for load balancing in hybrid networks in emerging markets by using four types of network topology.

This study can be used by network engineers and organizations as a foundation to start integrating SDN or plan a new SDN deployment to assist engineers to make quick decisions about optimal switch placement and SDN deployment through the hybrid approach to emerging markets.

8.4 Future Research Work

In future, the research can be extended to deploy a testbed to put innovations into consumer delivery one inch closer and the use of different vendor types of SDN switches. In load balancing, in order to accomplish more alternate transfers, the assessment need to use sizeable network configurations, including wider metros, to evaluate the load-balancing process. It is therefore recommended to incorporate the load balancer upgrade and seriously consider combined algorithms to even further boost the load balancing efficiency in hybrid SD-WAN ([Randhawa et al., 2018](#)). A more detailed study of the project planning and techno-economic analysis would form a basis for the initial stages of a migration process to a fully SDN African NREN environment and

expand our analysis to cover other African use-cases and consider scenarios where there already is SDN infrastructure. With the aim of improving switch placement, it would also be wise to explore open flow switch software testing ([Mimidis-Kentis et al., 2018](#)) and memory limitations of the SDN devices ([Marsico et al., 2017](#)) to assess how each manufacture's software affects the overall performance when a testbed is used. The switch to controller communication is important to improve network performance as well. As it is known that distributed controllers can delegate independent nodes to a master controller, in future, the study can look into solving the switch-controller problem altogether by using distributed controllers in WAN ([Kim et al., 2018](#)),

Bibliography

Abe, R., Miyamoto, S., Endo, Y. and Hamasuna, Y. (2017), Hierarchical clustering algorithms with automatic estimation of the number of clusters, *in* '2017 Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS)', IEEE, pp. 1–5.

Aerohive (2017), SD-LAN: THE FUTURE OF LOCAL AREA NETWORKS, Technical report.

URL: <https://i.crn.com/sites/default/files/ckfinderimages/userfiles/images/crn/custom/Aerohive-eBook-SDLAN.pdf>

Alliance, U. (2014), 'What is ubuntunet alliance', *Retrieved April 25, 2015*.

Alrashedy, K., Kimmett, B. and Gulliver, T. A. (2017), Performance of software-defined networking controllers for different network topologies, *in* 'IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)'.

Alsaeedi, M., Mohamad, M. M. and Al-Roubaiey, A. A. (2019), 'Toward adaptive and scalable openflow-sdn flow control: A survey', *IEEE Access* **7**, 107346–107379.

Amin, R., Reisslein, M. and Shah, N. (2018), 'Hybrid sdn networks: A survey of existing approaches', *IEEE Communications Surveys & Tutorials* **20**(4), 3259–3306.

Ardalan, K. (2012), 'Payback period and npv: their different cash flows', *Journal of economics and finance education* **11**(2).

Asadollahi, S., Goswami, B. and Sameer, M. (2018), Ryu controller's scalability experiment on software defined networks, *in* 'Current Trends in Advanced Computing (ICCTAC), 2018 IEEE International Conference on', IEEE, pp. 1–5.

Askar, S. K. (2016), 'Adaptive load balancing scheme for data center networks using software defined network', *Science Journal of University of Zakho* **4**(2), 275–286.

Badotra, S. and Singh, J. (2017), 'Open daylight as a controller for software defined networking.', *International Journal of Advanced Research in Computer Science* **8**(5).

- Beckett-Camarata, J. (2009), 'Local government measurement and use of performance accounting and financial reporting data in planning and budgeting decision support: A multicase study', *Public Performance & Management Review* **33**(2), 255–265.
- Belkadi, O. and Laaziz, Y. (2017), 'A systematic and generic method for choosing a sdn controller', *International Journal of Computer Networks and Communications Security* **5**(11), 239–247.
- Benabbou, J., Elbaamrani, K. and Idboufker, N. (2019), 'Security in openflow-based sdn, opportunities and challenges', *Photonic Network Communications* **37**(1), 1–23.
- Bhadana, A. and Singh, M. (2017), Fusion of k-means algorithm with dunn's index for improved clustering, in '2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)', IEEE, pp. 1–5.
- Bhatt, M. C. et al. (2018), Performance evaluation of load balancing algorithms in hadoop, in '2018 Second International Conference on Computing Methodologies and Communication (ICCMC)', IEEE, pp. 491–496.
- Bispo, P., Corujo, D. and Aguiar, R. L. (2017), A qualitative and quantitative assessment of sdn controllers, in '2017 International Young Engineers Forum (YEF-ECE)', IEEE, pp. 6–11.
- Boucadair, M. and Jacquenet, C. (2014), 'Software-defined networking: A perspective from within a service provider environment'.
- Caria, M., Jukan, A. and Hoffmann, M. (2013), A performance study of network migration to sdn-enabled traffic engineering, in '2013 IEEE Global Communications Conference (GLOBECOM)', IEEE, pp. 1391–1396.
- Casey, D. J. and Mullins, B. E. (2015), Sdn shim: Controlling legacy devices, in '2015 IEEE 40th Conference on Local Computer Networks (LCN)', IEEE, pp. 169–172.
- cbench* (n.d.).
URL: <https://github.com/mininet/oflops/tree/master/cbench>
- Chan, A., Keoleian, G. and Gabler, E. (2008), 'Evaluation of life-cycle cost analysis practices used by the michigan department of transportation', *Journal of Transportation Engineering* **134**(6), 236–245.
- Chembe, C., Kunda, D. and Simfukwe, M. (2014), 'Challenges and benefits of educational roaming (eduroam) service to zamren member institutions'.

- Chen, M.-H., Wang, W.-M., Chung, I.-H. and Chou, C.-F. (2017), Incremental hybrid sdn deployment for enterprise networks, *in* '2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)', IEEE, pp. 1143–1149.
- Corici, M., Reichel, B., Bochow, B. and Magedanz, T. (2016), An sdn-based solution for increasing flexibility and reliability of dedicated network environments, *in* '2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)', IEEE, pp. 1–6.
- Cui, Z., Liao, J., Wang, J., Qi, Q. and Wang, J. (2016), An approach to improve the cooperation between heterogeneous sdn overlays, *in* '2016 IEEE 41st Conference on Local Computer Networks (LCN)', IEEE, pp. 236–239.
- Dawadi, B. R., Rawat, D. B., Joshi, S. R. and Keitsch, M. M. (2018), Joint cost estimation approach for service provider legacy network migration to unified software defined ipv6 network, *in* '2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)', IEEE, pp. 372–379.
- Erlenback, J., Gutterman, F. J. and Tian, B. (2013), 'Method and apparatus for determining propagation delays for use in wide area networks'. US Patent 8,463,194.
- Eyisi, D. (2016), 'The usefulness of qualitative and quantitative approaches and methods in researching problem-solving ability in science education curriculum.', *Journal of Education and Practice* 7(15), 91–100.
- Fernando, M., Esteves, P., Esteve, C. et al. (2015), 'Software-defined networking: A comprehensive survey', *PROCEEDINGS OF THE IEEE* .
- Foley, M. (2016), 'The role and status of national research and education networks in africa'.
- Ganesh, J. S., Arisoğlu, M., Anand, P. and Sawant, N. M. (2017), 'Methods of forwarding data packets using transient tables and related load balancers'. US Patent 9,621,642.
- Gillwald, A. (2010), 'The poverty of ict policy, research, and practice in africa', *Information Technologies & International Development* 6(SE), pp–79.
- Gitman, L. J., Juchau, R. and Flanagan, J. (2015), *Principles of managerial finance*, Pearson Higher Education AU.

- Goransson, P., Black, C. and Culver, T. (2016), *Software defined networks: a comprehensive approach*, Morgan Kaufmann.
- Gu, B., Dong, M., Zhang, C., Liu, Z. and Tanaka, Y. (2017), Real-time pricing for on-demand bandwidth reservation in sdn-enabled networks, in '2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)', IEEE, pp. 696–699.
- Guo, A., Yuan, C., He, G. and Xu, L. (2018), Research on sdn/nfv network traffic management and optimization based on big data and artificial intelligence, in '2018 18th International Symposium on Communications and Information Technologies (ISCIT)', IEEE, pp. 377–382.
- Guo, Y., Wang, Z., Yin, X., Shi, X. and Wu, J. (2014), Traffic engineering in sdn/ospf hybrid network, in '2014 IEEE 22nd International Conference on Network Protocols', IEEE, pp. 563–568.
- Hakiri, A., Sellami, B., Patil, P., Berthou, P. and Gokhale, A. (2017), Managing wireless fog networks using software-defined networking, in '2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)', IEEE, pp. 1149–1156.
- Hamasuna, Y., Ozaki, R. and Endo, Y. (2017), A study on cluster validity measures for clustering network data, in '2017 Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS)', IEEE, pp. 1–6.
- Hayashi, N., Nojima, Y., Sasazuka, T., Yanagisawa, M. and Udagawa, A. (2017), 'Study on spillage of vomit for the purpose of preventing the spread of norovirus infection in hospitals and nursing homes', *Infectious Diseases Journal* **91**(3), 399–404.
- Horvath, R., Nedbal, D. and Stieninger, M. (2015), 'A literature review on challenges and effects of software defined networking', *Procedia Computer Science* **64**, 552–561.
- Hu, T., Yi, P., Zhang, J. and Lan, J. (2018), 'Reliable and load balance-aware multi-controller deployment in sdn', *China Communications* **15**(11), 184–198.
- Huang, M. and Liang, W. (2017), Incremental sdn-enabled switch deployment for hybrid software-defined networks, in '2017 26th International Conference on Computer Communication and Networks (ICCCN)', IEEE, pp. 1–6.
- Huang, X., Cheng, S., Cao, K., Cong, P., Wei, T. and Hu, S. (2018), 'A survey of deployment solutions and optimization strategies for hybrid sdn networks', *IEEE Communications Surveys & Tutorials* **21**(2), 1483–1507.
- Imt, I.-t. F. G. (2017), 'IMT-2020'.

Index, C. V. N. (2017), 'Forecast and methodology, 2016–2021', *White paper, Cisco public* **6**.

Jajoo, D. (2014), 'A study of buying decision process in malls'.

Jia, X., Jiang, Y. and Guo, Z. (2016), Incremental switch deployment for hybrid software-defined networks, in '2016 IEEE 41st Conference on Local Computer Networks (LCN)', IEEE, pp. 571–574.

Jones, C. (2004), 'Quantitative and qualitative research: conflicting paradigms or perfect partners?'.

Kar, B., Wu, E. H.-K. and Lin, Y.-D. (2016), 'The budgeted maximum coverage problem in partially deployed software defined networks', *IEEE Transactions on Network and Service Management* **13**(3), 394–406.

Karakus, M. and Durrezi, A. (2018), Economic impact analysis of control plane architectures in software defined networking (sdn), in '2018 IEEE International Conference on Communications (ICC)', IEEE, pp. 1–6.

Karakus, M. and Durrezi, A. (2019), 'An economic framework for analysis of network architectures: Sdn and mpls cases', *Journal of Network and Computer Applications* **136**, 132–146.

Kim, T., Choi, S.-G., Myung, J. and Lim, C.-G. (2017a), Load balancing on distributed datastore in opendaylight sdn controller cluster, in '2017 IEEE Conference on Network Softwarization (NetSoft)', IEEE, pp. 1–3.

Kim, T., Choi, S. G., Myung, J. and Lim, C. G. (2017b), 'Load balancing on distributed datastore in opendaylight SDN controller cluster', *2017 IEEE Conference on Network Softwarization: Softwarization Sustaining a Hyper-Connected World: en Route to 5G, NetSoft 2017*.

Kim, W., Li, J., Hong, J. W.-K. and Suh, Y.-J. (2018), 'Hes-cop: Heuristic switch-controller placement scheme for distributed sdn controllers in data center networks', *International Journal of Network Management* **28**(3), e2015.

Kouicem, D. E., Fajjari, I. and Aitsaadi, N. (2017), An enhanced path computation for wide area networks based on software defined networking, in '2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)', IEEE, pp. 664–667.

Kuang, H., Qiu, Y., Li, R. and Liu, X. (2018), A hierarchical k-means algorithm for controller placement in sdn-based wan architecture, in '2018 10th International

- Conference on Measuring Technology and Mechatronics Automation (ICMTMA)', IEEE, pp. 263–267.
- Lam, J., Lee, S.-G., Lee, H.-J. and Eko Oktian, Y. (2018), 'Design, implementation, and performance evaluation of identity-based cryptography in ONOS', *International Journal of Network Management* **28**(1), e1990.
- Lee, A., Wang, X., Nguyen, H. and Ra, I. (2018), 'A hybrid software defined networking architecture for next-generation iots.', *KSII Transactions on Internet & Information Systems* **12**(2).
- Levin, D., Canini, M., Schmid, S., Schaffert, F. and Feldmann, A. (2014), Panopticon: Reaping the benefits of incremental {SDN} deployment in enterprise networks, in '2014 {USENIX} Annual Technical Conference ({USENIX}{ATC} 14)', pp. 333–345.
- Mamushiane, L., Lysko, A. A. and Dlamini, S. (2018a), SDN-enabled infrastructure sharing in emerging markets: Capex/opex savings overview and quantification, in '2018 IST-Africa Week Conference (IST-AFRICA)', IEEE, pp. Page–1.
- Mamushiane, L., Lysko, A. A. and Mwangama, J. (2019), Resilient SDN controller placement optimization applied to and emulated on south african national research network (sanren), in 'submitted for Wireless Communications and Networking Conference', IEEE.
- Mamushiane, L., Lysko, A. and Dlamini, S. (2018b), A comparative evaluation of the performance of popular SDN controllers, in 'Wireless Days (WD), 2018', IEEE, pp. 54–59.
- Mamushiane, L., Lysko, A. and Mwangama, J. (2018), Optimum placement of SDN controllers in african backbones: SANREN and ZAMREN as a case study, in 'Southern Africa Telecommunication Networks and Applications Conference (SATNAC), 2018'.
- Mamushiane, L., Mwangama, J. and Lysko, A. A. (2018), Optimum placement of sdn controllers in african backbones: Sanren and zamren as a case study, SATNAC.
- Manel, M., Habib, Y. et al. (2017), An efficient mpls-based source routing scheme in software-defined wide area networks (sd-wan), in '2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)', IEEE, pp. 1205–1211.
- Marcus, J. S. and Molnar, G. (2017), 'Network sharing and 5g in europe: The potential benefits of using sdn or nfv'.
- Marsico, A., Doriguzzi-Corin, R. and Siracusa, D. (2017), An effective swapping mechanism to overcome the memory limitation of sdn devices, in '2017 IFIP/IEEE

- Symposium on Integrated Network and Service Management (IM)', IEEE, pp. 247–254.
- Michel, O. and Keller, E. (2017), Sdn in wide-area networks: A survey, *in* '2017 Fourth International Conference on Software Defined Systems (SDS)', IEEE, pp. 37–42.
- Mijumbi, R., Serrat, J., Gorricho, J.-L., Latré, S., Charalambides, M. and Lopez, D. (2016), 'Management and orchestration challenges in network functions virtualization', *IEEE Communications Magazine* **54**(1), 98–105.
- Mimidis-Kentis, A., Pilimon, A., Soler, J., Berger, M. and Ruepp, S. (2018), A novel algorithm for flow-rule placement in sdn switches, *in* '2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)', IEEE, pp. 1–9.
- Mitchell, D. W. (n.d.), 'From mpls to software-defined wide area network'.
- Nassaji, H. (2015), 'Qualitative and descriptive research: Data type versus data analysis'.
- Nkosi, M. C., Lysko, A. A. and Dlamini, S. (2018), Multi-path load balancing for sdn dataplane, *in* '2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC)', IEEE, pp. 1–6.
- ONF (2015), 'ONF TS-025 OpenFlow Switch Specification', **V 1.5.1**, 1–283.
- ONF, O. O. S. (2012), 'v1. 3.0. jun. 2012', *ONF Specification* .
- Ozaki, R., Hamasuna, Y. and Endo, Y. (2017), Agglomerative hierarchical clustering based on local optimization for cluster validity measures, *in* '2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)', IEEE, pp. 1822–1827.
- Patrushev, G. G. and Drozdova, V. G. (2017), Routing efficiency evaluation with sdn solutions integration in the data network, *in* '2017 18th International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices (EDM)', IEEE, pp. 190–194.
- Pit-Claudiel, B., Desmouceaux, Y., Pfister, P., Townsley, M. and Clausen, T. (2018), Stateless load-aware load balancing in p4, *in* '2018 IEEE 26th International Conference on Network Protocols (ICNP)', IEEE, pp. 418–423.
- Randhawa, N. S., Dhimi, M. and Singh, P. (2018), Enhanced load balancer with multilayer processing architecture for heavy load over cloud network, *in* 'Applications of Data Management and Analysis', Springer, pp. 169–176.

- Rathee, S., Dinesh Ram Kumar, T., Haribabu, K. and Bhatia, A. (2019), A framework to achieve full waypoint enforcement in hybrid sdn networks, *in* L. Barolli, M. Takizawa, F. Xhafa and T. Enokido, eds, 'Advanced Information Networking and Applications', Springer International Publishing, Cham, pp. 325–340.
- Risdianto, A. C., Tsai, P.-W., Ling, T. C., Yang, C.-S. and Kim, J. (2017), 'Enhanced onos sdn controllers deployment for federated multi-domain sdn-cloud with sd-routing-exchange', *Malaysian Journal of Computer Science* **30**(2), 134–153.
- Rojas, E., Doriguzzi-Corin, R., Tamurejo, S., Beato, A., Schwabe, A., Phemius, K. and Guerrero, C. (2018), 'Are we ready to drive software-defined networks? a comprehensive survey on management tools and techniques', *ACM Computing Surveys (CSUR)* **51**(2), 27.
- Rosendo, D., Endo, P. T., Sadok, D. and Kelner, J. (2017), An autonomic and policy-based authorization framework for openflow networks, *in* '2017 13th International Conference on Network and Service Management (CNSM)', IEEE, pp. 1–5.
- SANReN CA, T. (2018), 'The South African NREN – SANReN'.
URL: <http://www.sanren.ac.za/south-african-nren/>
- Santacreu, A. M. (2015), 'The economic fundamentals of emerging market volatility', *Economic Synopses* (2).
- Šeremet, I. and Čaušević, S. (n.d.), 'Evolving ip/mpls network in order to meet 5g requirements'.
- Shozi, T., Dlamini, S., Mudali, P. and Adigun, M. O. (2019), An sdn solution for performance improvement in dedicated wide-area networks, *in* '2019 Conference on Information Communications Technology and Society (ICTAS)', IEEE, pp. 1–6.
- Singh, A. K. and Srivastava, S. (2018), 'A survey and classification of controller placement problem in SDN', *International Journal of Network Management* **28**(3), e2018.
- Sinha, Y., Haribabu, K. et al. (2017), 'A survey: Hybrid sdn', *Journal of Network and Computer Applications* **100**, 35–55.
- Takamura, T., Sakurai, H., Yamamoto, D., Murakami, M., Yamasaki, K. and Unno, Y. (2019), 'Realizing next-generation hybrid networks for multi-cloud era', *FUJITSU SCIENTIFIC & TECHNICAL JOURNAL* **55**(3), 61–68.
- Tello-Oquendo, L., Akyildiz, I. F., Lin, S.-C. and Pla, V. (2018), Sdn-based architecture for providing reliable internet of things connectivity in 5g systems, *in* '2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)', IEEE, pp. 1–8.

- Thyagaturu, A. S., Mercian, A., McGarry, M. P., Reisslein, M. and Kellerer, W. (2016), 'Software defined optical networks (sdons): A comprehensive survey', *IEEE Communications Surveys & Tutorials* **18**(4), 2738–2786.
- Trajkowska, I., Kourtis, M.-A., Sakkas, C., Baudinot, D., Silva, J., Harsh, P., Xylouris, G., Bohnert, T. M. and Koumaras, H. (2017), 'Sdn-based service function chaining mechanism and service prototype implementation in nfvs scenario', *Computer Standards & Interfaces* **54**, 247–265.
- Vengainathan, B., Basil, A., Tassinari, M., Manral, V. and Banks, S. (2018), Benchmarking methodology for sdn controller performance, Technical report, Internet-Draft draft-ietf-bmwg-sdn-controller-benchmark-meth-08, Internet
- Ventre, P. L., Ortiz, J., Mendiola, A., Fernández, C., Pavlidis, A., Sharma, P., Buscaglione, S., Stamos, K., Sevasti, A. and Whittaker, D. (2017), Deploying sdn in geant production network, in '2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)', IEEE, pp. 1–2.
- Virtualisation, N. F. (2012), An introduction, benefits, enablers, challenges & call for action, in 'White Paper, SDN and OpenFlow World Congress'.
- Vissicchio, S., Cittadini, L., Vissicchio, S. and Cittadini, L. (2017), 'Safe, efficient, and robust sdn updates by combining rule replacements and additions', *IEEE/ACM Transactions on Networking (TON)* **25**(5), 3102–3115.
- Vissicchio, S., Vanbever, L. and Bonaventure, O. (2014), 'Opportunities and research challenges of hybrid software defined networks', *ACM SIGCOMM Computer Communication Review* **44**(2), 70–75.
- Wang, D. W. (2018), *Software Defined-WAN for the Digital Age: A Bold Transition to Next Generation Networking*, CRC Press.
- Wang, G., Zhao, Y., Huang, J., Duan, Q. and Li, J. (2016), A k-means-based network partition algorithm for controller placement in software defined network, in '2016 IEEE International Conference on Communications (ICC)', IEEE, pp. 1–6.
- Wang, G., Zhao, Y., Huang, J. and Wu, Y. (2017), 'An effective approach to controller placement in software defined wide area networks', *IEEE Transactions on Network and Service Management* **15**(1), 344–355.
- Wara, Y. and Singh, D. (2015), 'A guide to establishing computer security incident response team (csirt) for national research and education network (nren)', *African Journal of Computing & ICT* **8**(2), 1–8.

- Xu, H., Huang, H., Chen, S., Zhao, G. and Huang, L. (2018), 'Achieving high scalability through hybrid switching in software-defined networking', *IEEE/ACM Transactions on Networking* **26**(1), 618–632.
- Zaidi, Z., Friderikos, V., Yousaf, Z., Fletcher, S., Dohler, M. and Aghvami, H. (2018), 'Will sdn be part of 5g?', *IEEE Communications Surveys & Tutorials* **20**(4), 3220–3258.
- Zakia, U. and Yedder, H. B. (2017), Dynamic load balancing in sdn-based data center networks, in '2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)', IEEE, pp. 242–247.
- Zarin, N. and Agarwal, A. (2018), A centralized approach for load balancing in heterogeneous wireless access network, in '2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)', IEEE, pp. 1–5.
- Zhou, S., Xu, Z. and Liu, F. (2016), 'Method for determining the optimal number of clusters based on agglomerative hierarchical clustering', *IEEE transactions on neural networks and learning systems* **28**(12), 3007–3017.
- Zhou, Y., Wang, Y., Yu, J., Ba, J. and Zhang, S. (2017), Load balancing for multiple controllers in sdn based on switches group, in '2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS)', IEEE, pp. 227–230.

Appendices

A Raw Data from Simulations

Table A1 presents the latency (i.e. round-trip time (RTT)) results obtained from our emulation experiments when one ONOS controller is deployed.

Table A1: Total average latency for one controller

Switch ID	Region Name	RTT (ms)
s1	Johannesburg	107.58
s2	Pretoria	95.62
s3	Durban	91.87
s4	Bloemfontein	164.4
s5	East London	121.63
s6	Port Elizabeth	106.68
s7	Cape Town	88.78

Table A2 presents the latency (i.e. round-trip time (RTT)) results obtained from our emulation experiments when two ONOS controllers are deployed.

Table A2: Total average latency for two controllers

Switch ID	Region Name	RTT (ms)
s2 & s1	Pretoria & Johannesburg	52.316
s2 & s5	Pretoria & East London	48.933
s2 & s6	Pretoria & Port Elizabeth	49.95
s2 & s7	Pretoria & Cape Town	52.23
s3 & s1	Durban & Johannesburg	51
s3 & s5	Durban & East London	49.62
s3 & s6	Durban & Port Elizabeth	53.62
s3 & s7	Durban & Cape Town	56.87
s4 & s1	Bloemfontein & Johannesburg	66.02
s4 & s5	Bloemfontein & East London	68.13
s4 & s6	Bloemfontein & Port Elizabeth	118.35
s4 & s7	Bloemfontein & Cape Town	112.67

Table [A3](#) presents the results obtained from our experiments regarding the impact of switch-to-controller placement balancing on SDN controller performance. The key performance indicators considered in these experiments are controller average delay, average packet loss and jitter.

Table A3: Effect of switch placement on controller performance

No. of Packets	Before switch-to-controller placement balancing			After switch-to-controller placement balancing		
	% Packet loss	Average Delay	Average Jitter	% Packet loss	Average Delay	Average Jitter
50 000	0	0.057	0.023	0	0.049	0.025
100 000	0	0.026	0.01	0	0.017	0.004
150 000	0.19	0.015	0.004	0.14	0.019	0.006
200 000	0.53	0.026	0.005	0.07	0.024	0.008

Tables [A4](#) and [A5](#) present the results obtained from our controller benchmarking experiments. The results in Table [A4](#) were obtained by varying the number of virtual switches under a fixed number of MACs (i.e. Number of MACs = 1000). Table [A5](#) shows results obtained for a varying number of MACs whilst the number of switches is fixed at 16 switches

Table A4: Benchmarking results under a varying number of switches

No. of Switches	Throughput				Latency				Compound Performance			
	Floodlight	Ryu	ONOS	OpenDayLight	Floodlight	Ryu	ONOS	OpenDayLight	Floodlight	Ryu	ONOS	OpenDayLight
1	26051.21	5942.35	30407.51	43082.72	12.90	2.98	25.91	2.96	2018866.40	1988392.25	1173620.06	14534006.69
4	107236.75	5810.87	119699.82	98418.04	41.82	3.64	64.98	11.49	2564167.71	1595608.20	1842116.95	8565874.21
8	178396.00	5634.29	222293.52	63988.61	56.37	3.92	60.92	15.92	3164561.80	1437744.33	3649203.33	4019819.35
12	109641.47	5701.41	274439.70	56219.76	51.71	4.04	57.67	19.92	2120473.75	1412019.51	4758907.09	2822311.11
16	92692.79	5748.85	255122.81	55840.85	54.34	3.94	60.43	26.98	1705658.94	1458762.06	4221577.60	2070045.74
20	87245.83	5590.35	263027.37	53217.50	56.79	4.02	62.83	27.88	1536296.45	1392057.55	4186264.78	1908481.13
24	86658.10	5523.29	257741.71	48140.64	60.16	4.04	62.72	35.31	1440405.85	1367465.78	4109239.13	1363486.36
28	-	5357.17	253676.33	46904.44	60.16	4.22	64.76	24.61	-	1267726.19	3917061.27	1905740.20
32	-	5334.92	257037.10	43420.51	52.64	4.16	61.19	29.78	-	1283719.50	4200312.25	1457954.03

Table A5: Benchmarking results under a varying number of MACs

No. of MACs	Throughput				Latency				Compound Performance			
	Floodlight	Ryu	ONOS	OpenDayLight	Floodlight	Ryu	ONOS	OpenDayLight	Floodlight	Ryu	ONOS	OpenDayLight
1000	26051.21	5942.35	30407.51	32867.56	12.90	2.99	25.91	30.11	2018866.42	1988392.25	1173620.06	1091679.67
10 000	60919.16	6286.60	248668.61	33192.59	53.86	4.10	58.77	31.29	1130969.65	1534099.09	4231100.87	1060803.02
100 000	10521.75	6421.19	256598.40	31677.42	12.90	4.20	63.17	20.65	815799.53	1530483.63	4062212.72	1534037.78
1 000 000	12092.92	6331.94	262218.59	33298.36	12.72	4.35	63.70	22.48	950817.12	1454107.46	4116280.45	1481317.57
100 000 000	12568.29	6129.53	245473.89	29894.65	12.55	4.42	63.17	23.60	1001805.41	1388351.45	3885715.37	1266729.44