A logo with black text

Description automatically generated

**Literature Review**

A comprehensive analysis of traffic load balancers in software-defined networking.

Salon Ghalan Tamang

CST 3990

A logo with black text

Description automatically generated

**Table of Contents**

Chapter 1: Literature Review………………………...........................1

1.1 Introduction……………………………………………………………………1

1.2 Relevant Literature………………………………….………………………1

1.3 Conclusion……………………………………………….……………………..5

Chapter 2: Initial Steps………………………………….………………………6

2.1 Overview………………………………………………….……………………..6

2.2 System description…………………………………..………………………6

2.3 Technology stack……………………………………..………………………6

2.4 Network design………………………………………..……………………..7

**A logo with black text

Description automatically generated**

**Chapter 1: Literature Review**

* 1. **Introduction**

Software-defined network (SDN) is an emerging technology in computer networks. By partitioning the current network into a centralized control plane (CP) and a remotely programmable data plane (DP), SDN streamlines the design, control, and management of next-generation networks, such as 5G, cloud computing, and big data. A SDN southbound interface (SBI) links the DP and the CP (Lamiae Boukraa et al., 2022). Previously, in a traditional network, DP (the actual forwarding elements) and CP (the logic underlying the forwarding functionality) are packaged into a single box and connected tightly. The overall design remains complex and expensive due to its intricate and tight integration. In contrast to this, SDNs are flexible, dynamic, cost-effective, and customizable, making them ideal for the dynamic nature of today's applications (Rukmini Bhat B et al., 2021). However, the ever-growing avalanche of network traffic has offered many challenges to SDNs in terms of offering reliable connectivity, quality of service and scalability. Hence, load balancing (LB) has a crucial role in SDN. In this literature, I will be reviewing publications on several LB techniques that allow for dynamic traffic management and efficient utilization of network resources in SDN.

* 1. **Relevant Literature**

The rapid growth of the internet with advancements in communication technologies has flooded the networks all over the world. This increasing demand coupled with high-speed data transmission requirements needs LB and proper management of server resources. (Fakhrun Jamal & Tamanna Siddiqui, 2021) in their paper defines LB as the practice of reallocating load to various nodes of common infrastructure to develop resource proficiency and increase the job's answer period whereas similarly eliminating a condition in which some nodes are overloaded, and others are underloaded. In the journal, they have classified LB mechanisms into three categories namely process initiation based, system state-based and spatial distribution of node-based. They discovered that static LB is less efficient than dynamic LB, but dynamic LB is less productive than the hybrid technique, which incorporates QoS and performance requirements (Response time, Reliability, Resource utilization, fault tolerance, and Scalability).

**1**

A logo with black text

Description automatically generated

A table of text with black and white text

Description automatically generated

Figure 1: comparison table of LB techniques from Fakhrun Jamal & Tamanna Siddiqui, 2021.

Another study by (Fancy et al., 2019) compared traditional networks with SDN networks. The SDN networks have shown great advantages in many aspects, but if the load distribution is uneven in the SDN networks, it will greatly affect the performance of the network. The three load balancing algorithms considered for analysis purpose are least connection, Round Robin and Weighted Round Robin algorithm. Based on throughput analysis, it showed that round robin algorithm could only provide the least number of transactions. Also, it is evident that the least connection algorithm (a dynamic LB algorithm) provides the highest throughput when compared to the Weighted Round Robin and Round Robin algorithms (static LBs). Furthermore, a recent study by (Rukmini Bhat B et al., 2021) shows how a derived method using least connection algorithm with Dijkstra’s algorithm could be used in a single SDN controller to reduce round trip time (RTT) in overall network. The purposed solution claims that the average standard deviation for sending 200 packets 10 number of times reduced from 0.9449ms to 0.429 after applying the algorithm. However, this solution failed to consider traffic volume in real time.

A solution considering traffic volume and distributing traffic equally to avoid data congestion and link overloading has been proposed by (M.C. Nkosi, A.A. Lysko, S. Dlamini, 2018). The proposed system uses multipath-based LB in a single centralized Opendaylight controller’s data

**2**

A logo with black text

Description automatically generated

plane. This LB algorithm takes source and destination as input and calculates alternative short paths. Moving on, the paths are pushed down to the flow table. The algorithm then computes path costs for all defined paths using network statistics and load balancing is repeated until cost for all paths are equal. As a result, the load balancer showed improved network performance in transfer rate and response time. However, it was found that for better network improvement, the data plane should have multiple alternative links so that multiple paths can be defined for a routing path. Conventional networks use centralized controllers to coordinate traffic distribution. In a study by (P. Dharam and M. Dey, 2021) mentions that a single controller can no longer support the application requirements of SDN networks due to the growth of network applications and continuing scale expansion. Additionally, a single controller is vulnerable to failure issues and overload, which is detrimental to the scalability and reliability of networks. Hence, large SDN networks with numerous controllers in the control plane have emerged. Each controller oversees a group of switches, and the SDN network is managed by a control plane that is both physically and logically distributed.

(Kai-Yu Wang, Shang-Juh Kao and Ming-Tsung Kao, 2018) in their paper proposes a solution to challenges of using a single dedicated controller. They implemented a distributed approach with multiple controllers to avoid a single point of failure. Their proposed system focuses on balancing load across multiple controllers and shifts packet flows to a controller with a lighter load. In the proposed system, three logical components were included in each controller, namely a load collector, which periodically activates to collect loading statuses from other collectors and share its loading information, a load balancer which records its identification, loading information, associated switches and most recent update time and a switch migrator, which notifies its associated switch to redirect the forwarding path to the target controller once load of a controller exceeds the threshold. Moreover, to avoid simultaneous migration assignment to same controller, the balancer module processes information of one controller at a time and the controller’s sequent actions were delayed for 5 seconds when an overloaded controller registered another overloaded controller on its load record. The outcome of this proposed system confirmed that the standby controller solved the reliability problem. But delaying controller actions for a brief period involves a tradeoff of bandwidth conscription. In addition, the controller has predefined static threshold which could not align parallel with network changes and, the controller assignment does not consider the dynamic loads of network.

(Songzhou Li et al, 2023) outlines the problems associated with fixed threshold setting and introduces controller LB algorithm based on dynamic threshold (CLBDT), which has LB and threshold adjustment functions. The experiment is simulated in Mininet using Floodlight

**3**

A logo with black text

Description automatically generated

controller and Cbench were used to simulate flow requests from the switch to send Packet-In messages to controller. From the simulation results, they found that the load standard deviation of the fixed threshold load balancing algorithm is calculated to be 0.098 and the load standard deviation of the CLBDT algorithm is 0.010. Hence, they showed that CLBDT algorithm has better load balancing effect than fixed threshold LB algorithm.

Another research paper by (Zhihao Shang et al, 2019) illustrates the dynamic controller assignment problem as NP-complete problem. In their proposed solution, they designed a heuristic for solving the controller assignment problem named LANS (Late Acceptance Neighbor Search). It accepts the current controller assignment matrix and provides a new controller assignment matrix that can minimize the weighted sum of the flow setup time and switch migration time. They used a greedy algorithm that generates a feasible controller assignment matrix which is later fed to LANS. This algorithm sorts the controllers by their utilizations and tries to balance load in the controller with least migrations. They compared their solution with dynamic controller assignment algorithm DCP-SA in terms of flow setup time, migrations, and CV (coefficient of variation). They measured the flow setup time of a controller and modeled each controller as an *M/PH/1* queue to capture its performance. The queueing model is used in the heuristic for the fitness function. The results showed that their solution can balance the controller better, reduce the flow setup time and make less migrations less than DCP-SA.

Controller failure is a critical challenge in distributed SDN. A research paper by (Poonam Dharam & Mithila Dey, 2021) implements two dynamic solutions namely Random Weight Load Balancing and Progressive Assignment Load Balancing discarding proactively assigned pre-partition which do not consider current network state. Their solution setup consists of multiple controllers and a LB. The LB keeps track of global topology network and has a monitoring module that periodically sends heartbeat messages to all connected controllers. Results from their simulation showed that proposed solutions successfully reassigned all the orphan switches to other active controllers such that the load of controllers after assignment is close to each other. Nevertheless, it is important to note that the LB is a single entity that handles the critical task of assigning orphan switches to available controllers. Hence, failure of such centralized component would freeze the network in turn affecting QoS.

Finding the shortest path between source and destination is detrimental in maintaining performance of distributed SDN. In a recent paper by (Dmitry P. et al, 2023) proposes an intelligent multipath routing method based on artificial neutral network that allows the controller to configure data transmission policies quickly and efficiently. Hyperparameters when designing the neural network model are optimized by artificial bee colony algorithm. The accuracy of the model predicting the shortest paths is about 90%. However, the authors fail to

**4**

A logo with black text

Description automatically generated

show integration of this solution in SDN controller with simulation making real-time routing decisions.

**1.3 Conclusion**

**­­­­­**The purpose of this review was to analyze recent research papers on traffic load balancing in SDN. Most research papers gathered show that dynamic distributed approaches have superior performance over static load balancing algorithms. Based on various studies considered, multiple techniques still lack consideration of various performance metrics. Nevertheless, more exploration of hybrid LB balancing techniques should be encouraged. Furthermore, as technology is advancing and networks are congesting on daily basis, I believe that extensive research on load balancing schemes integrating artificial intelligence is also required to refine reliability, scalability, and efficiency of SDN in the future.

**5**

A logo with black text

Description automatically generated

**Chapter 2: Initial Steps**

**2.1. Overview**

In this chapter, I will be walking you through the steps talking towards building the simulation network and current stage of the project. This chapter will highlight the requirements for implementation of simulation software. This chapter will also showcase the network topology design and algorithm used for load balancing of the SDN.

**2.2. System description**

Since, this is going to be analysis report for traffic load balancers, I will be using different LB techniques for comparison. Two of them are round-robbin (a static LB algorithm) and least connection with Dijkstra's algorithm (dynamic LB method). Among many controllers such as Floodlight, POX, Ryu, ONOS etc, I will be using Opendaylight controller and Mininet, a network emulator that creates a network for implementation.

**2.3 Technology stack**

* **Mininet** – a network simulation software
* **Opendaylight Controller (Oxgyen version)** – a SDN controller
* **Ubuntu 22.04.3 LTS –** the operating system used for implementation.
* **Python -** programming language for SDN controller’s LB algorithm
* **DLUX –** a graphical user interface for Opendaylight controller
* **Cbench** - a software to simulate flow requests from the switch to send Packet-In messages to controller.

**6**

**2.4 Network design**

Python was used to create a topology. In the topology, we have 3 switches, 9 hosts and 1 centralized controller. At first, all required libraries and methods were imported.

A screenshot of a computer program

Description automatically generated

Fig 2: Importing required method from mininet in python

A screen shot of a computer

Description automatically generated

Fig 3: Part 1 of create\_topology method

**7**

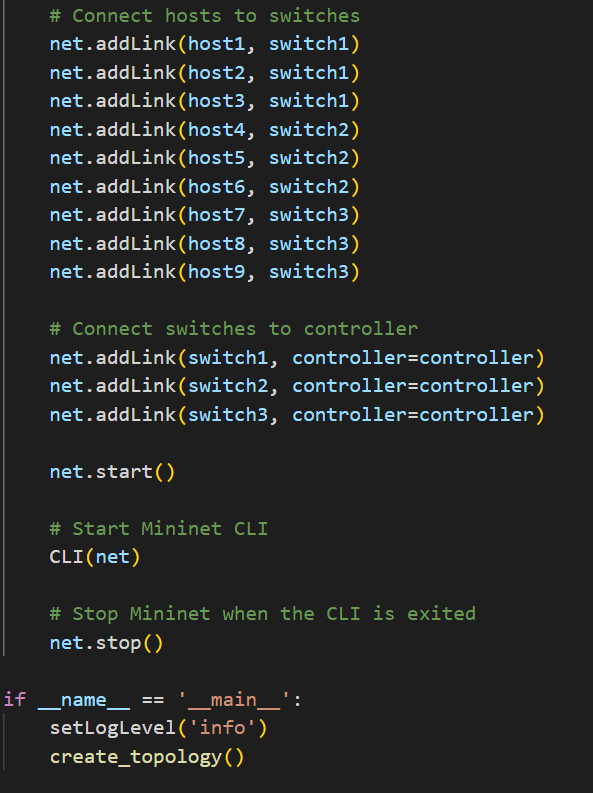


Fig 4: part 2 of create\_topology method

**8**

A logo with black text

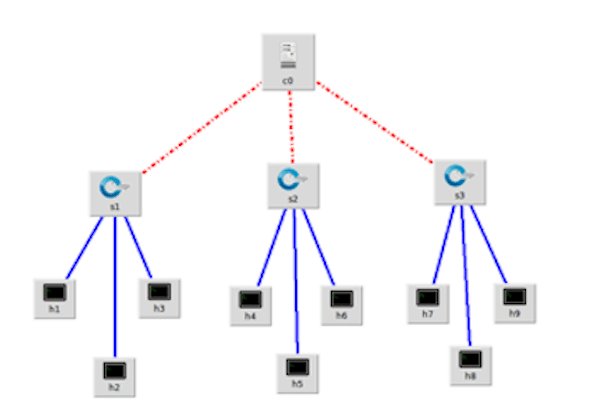
Description automatically generated

Fig 5: Network topology

**9**

A logo with black text

Description automatically generated

**Research Ethics Screening Form for Students**

Middlesex University is concerned with protecting the rights, health, safety, dignity, and privacy of its research participants. It is also concerned with protecting the health, safety, rights, and academic freedom of its students and with safeguarding its own reputation for conducting high quality, ethical research.

*This Research Ethics Screening Form will enable students to self-assess and determine whether the research requires ethical review and approval via the Middlesex Online Research Ethics (MORE) form before commencing the study. Supervisors must approve this form after consultation with students.*

|  |  |  |
| --- | --- | --- |
| Student Name: | Salon Ghalan Tamang | Email: sg1591@live.mdx.ac.uk |
| Research project title: | A comprehensive analysis of traffic load balancing in software-defined networking | |
| Programme of study/module: | BSc Computer Science / CST3990 Undergraduate Individual Project | |
| Supervisor Name: | Clifford Sule | Email: c.sule@mdx.ac.uk |

|  |  |  |
| --- | --- | --- |
| *Please answer whether your research/study involves any of the following given below:* | | |
| 1. HANIMALS or animal parts. | Yes | No |
| 1. MCELL LINES (established and commercially available cells - biological research). | Yes | No |
| 1. HCELL CULTURE (Primary: from animal/human cells- biological research). | Yes | No |
| 1. HCLINICAL Audits or Assessments (e.g. in medical settings). | Yes | No |
| 1. XCONFLICT of INTEREST or lack of IMPARTIALITY.  If unsure see “Code of Practice for Research” (Sec 3.5) at:  https://unihub.mdx.ac.uk/study/spotlights/types/research-at-middlesex/research-ethics | Yes | No |
| 1. XDATA to be used that is not freely available (e.g. secondary data needing permission for access or use). | Yes | No |
| 1. XDAMAGE (e.g., to precious artefacts or to the environment) or present a significant risk to society). | Yes | No |
| 1. XEXTERNAL ORGANISATION – research carried out within an external organisation or your reseach is commissioned by a government (or government body). | Yes | No |
| 1. MFIELDWORK (e.g biological research, ethnography studies). | Yes | No |
| 1. HGENETICALLTY MODIFIED ORGANISMS (GMOs) (biological research). | Yes | No |
| 1. HGENE THERAPY including DNA sequenced data (biological research). | Yes | No |
| 1. MHUMAN PARTICIPANTS – ANONYMOUS Questionnaires (participants not identified or identifiable). | Yes | No |
| 1. XHUMAN PARTICIPANTS – IDENTIFIABLE (participants are identified or can be identified): survey questionnaire/ INTERVIEWS /  focus groups / experiments / observation studies. | Yes | No |
| 1. HHUMAN TISSUE (e.g., human relevant material, e.g., blood, saliva, urine, breast milk, faecal material). | Yes | No |
| 1. HILLEGAL/HARMFUL activities research (e.g., development of technology intended to be used in an illegal/harmful context or to breach security systems, searching the internet for information on highly sensitive topics such as child and extreme pornography, terrorism, use of the DARK WEB, research harmful to national security). | Yes | No |
| 1. XPERMISSION is required to access premises or research participants. | Yes | No |
| 1. XPERSONAL DATA PROCESSING (Any activity with data that can directly or indirectly identify a living person).  For example data gathered from interviews, databases, digital devices such as mobile phones, social media or internet  platforms or apps with or without individuals'/owners' knowledge or consent, and/or could lead to individuals/owners being IDENTIFIED or SPECIAL CATEGORY DATA  (GDPR1) or CRIMINAL OFFENCE DATA.   1Special category data (GDPR- Art.9): ”personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership, and the processing of genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person’s sex life or sexual orientation”. | Yes | No |
| 1. XPUBLIC WORKS DOCTORATES: Evidence of permission is required for use of works/artifacts (that are protected by Intellectual Property (IP) Rights, e.g. copyright, design right) in a doctoral critical commentary when the IP in the work/artifactis jointly prepared/produced or is owned by another body | Yes | No |
| 1. HRISK OF PHYSICAL OR PSYCHOLOGICAL HARM (e.g., TRAVEL to dangerous places in your own country or in a foreign country (see <https://www.gov.uk/foreign-travel-advice>), research with NGOs/humanitarian groups in conflict/dangerous zones, development of technology/agent/chemical that may be harmful to others, any other foreseeable dangerous risks). | Yes | No |
| 1. XSECURITY CLEARANCE – required for research. | Yes | No |
| 1. XSENSITIVE TOPICS (e.g., anything deeply personal and distressing, taboo, intrusive, stigmatising, sexual in nature, potentially dangerous, etc). | Yes | No |

M – Minimal Risk; X – More than Minimal Risk. H – High Risk

If you have answered 'Yes' to ANY of the items in the table, your application REQUIRES ethical review and approval using the MOREform **BEFORE commencing your research**. Please apply for ethical approval using the MOREform (<https://moreform.mdx.ac.uk/>). Consult your supervisor for guidance. Also see *Middlesex Online Research Ethics (*MyLearning area*) and* [*www.tiny.cc/mdx-ethics*](http://www.tiny.cc/mdx-ethics)(CS students).

If you have answered 'No' to ALL of the items in the table, your application is Low Risk and you may NOT require ethical review and approval using the MOREform before commencing your research. Your research supervisor will confirm this below.

A logo with black text

Description automatically generated

Student Signature: Salon Ghalan Tamang Date:10th January, 2024

**To be completed by the supervisor:**

|  |  |
| --- | --- |
| *Based on the details provided in the self-assesment form, I confirm that:* | Insert Y or N |
| The study is Low Risk and *does not require* ethical review & approval using the MOREform | Y |
| The study *requires* ethical review and approval using the MOREform. |  |

Superivsor Signature:…C.Sule….. Date:…11/01/2024……

**References**

Lamiae Boukraa, Safaa Mahrach, Khalid El Makkaoui & Redouane Esbai SDN (2022). Southbound Protocols: A Comparative Study. International Conference on Networking, Intelligent Systems and Security. [online] Available at: <https://link.springer.com/chapter/10.1007/978-3-031-15191-0_39#SnippetTab>

Rukmini Bhat B \*, Sneha N S, Keerthana Bhat, Chaithra C Kamath,Chaitha Naik (2021). Improving the Efficiency of Software Defined Network through Load Balancing Algorithms. Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV). [online] Available at: <https://ieeexplore.ieee.org/document/9388512>

Fakhrun Jamal, Tamanna Siddiqui (2021). Comparative Analysis of Load Balancing Techniques in Cloud Computing, Based on LB Metrices. 5th International Conference on Information Systems and Computer Networks (ISCON). [online] Available at: <https://ieeexplore.ieee.org/document/9702508>

C. Fancy, M. Pushpalatha, Pushpa (2019). Experimentation of Traditional Load Balancing Algorithms in Software Defined Network. International Journal of Recent Technology and Engineering (IJRTE), ISSN: 2277-3878, Volume-8, Issue-1S4, June 2019. [online] Available at: ijrte.org/wp-content/uploads/papers/v8i1s4/A10970681S419.pdf

M.C. Nkosi, A.A. Lysko, S. Dlamini (2018). Multi-path Load Balancing for SDN Data Plane. 2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC). [online] Available at: <https://ieeexplore.ieee.org/document/8601241>

A logo with black text

Description automatically generated

P. Dharam and M. Dey (2021). A Mechanism for Controller Failover in Distributed Software-Defined Networks. 2021 8th International Conference on Computer and Communication Engineering (ICCCE). [online] Available at: <https://ieeexplore.ieee.org/document/9467174>

Kai-Yu Wang, Shang-Juh Kao and Ming-Tsung Kao (2018). An Efficient Load Adjustment for Balancing Multiple Controllers in Reliable SDN Systems. Proceedings of IEEE International Conference on Applied System Innovation 2018. [online] Available at: <https://ieeexplore.ieee.org/document/8394323>

Songzhou Li, Zhenghua Xin, Xu Xu & Zhiwei Zhang (2023). Load Balancing Algorithm of SDN Controller Based on Dynamic Threshold. 2023 3rd Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS). [online] Available at: <https://ieeexplore.ieee.org/document/10145235>

Zhihao Shang, Han Wu, Guang Peng and Katinka Wolter (2019). Dynamic Load Balancing in the Control Plane of Software-Defined Networks. 2019 IEEE 19th International Conference on Communication Technology. [online] Available at: <https://ieeexplore.ieee.org/document/8947243>

Poonam Dharam, Mithila Dey (2021). A Mechanism for Controller Failover in Distributed Software-Defined Networks. 2021 8th International Conference on Computer and Communication Engineering (ICCCE). [online] Available at: <https://ieeexplore.ieee.org/document/9467174>

Dmitry Perepelkin, Maria Ivanchikova, Tin Nguyen (2023). Neural Network Multipath Routing in Software Defined Networks Based on Artificial Bee Colony Algorithm. 2023 XVIII International Symposium Problems of Redundancy in Information and Control Systems (REDUNDANCY). [online] Available at: https://ieeexplore.ieee.org/document/10330174

Alexandra Eftimie, Eugen Brococi (2020). SDN controller implementation using OpenDaylight: experiments. [2020 13th International Conference on Communications (COMM)](https://ieeexplore.ieee.org/xpl/conhome/9139890/proceeding). [online] Available at:SDN controller implementation using OpenDaylight: experiments