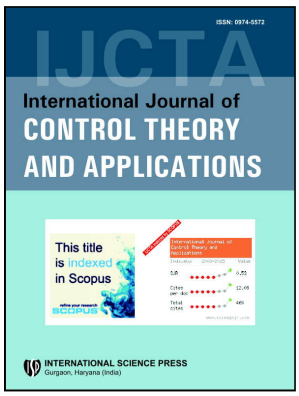

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 18 • 2017

Model of Congestion Control in Software-Defined Network

Mohini Singh[1], Karan Singh[2] and Shiva Prakash[3]

[1] *Master of Technology In Computer Science Banasathali Vidyapith, Jaipur*
[2] *Assistant Professor, School of Computer & Systems Sciences Jawaharlal Nehru University, New Delhi*
[3] *Associate Professor In Department of Computer Science & Engineering Madan Mohan Malaviya University of Technology, Gorakhpur*

Abstract: Our current environment moving real world to virtual world. Here various way to move real world to virtual world. NFV, SDN and more other way to change real world network to virtual world network. Our upcoming world is based on mostly virtualization. In our paper we discuss the Software Defined Network. We also describe congestion in Software Defined Network and how to reduce that congestion in Software Defined Network. We proposed the method to control the congestion in Software defined Network.

1. INTRODUCTION

The aim of Software-Defined Networking is making the network more flexible as the virtual server and collecting infrastructure of the modern data centers. There is different way to define the Software-Defined Networking.

In 90's century networks were establish for communication mainly. They were built a device running distributed protocols that are provide various functionality like access control, congestion control, security, error minimization , traffic monitoring. Software-Defined Networking is like as a container which has several type of networking technology.

In present time networking is not part of only communication. They are also used in the business, marketing, sharing, shopping and other things. They are not possible with the traditional network because of static storage and inflexibility.

In Fig.1. describe the architecture basis difference between Traditional network and Software-Defined Network. Traditional network is combination of the data plane and control plan, but Software Defined Network is separated architecture where control plane are centralized for all the data plane.

Traditional network has the static storage. Software Defined Network is dynamic in storage and flexible.Table 1. describe the difference between Traditional network and Software-Defined Network on different parameter.

Table 1
Difference Between The Traditional Network And Software-Defined Network

Parameter	*Traditional Network*	*Software-Defined Network*
Protocol	Not determine	OpenFlow
Venue	Service Provider Network	Data Center/ Cloud
Dynamic Structure	Complex	Quickly adapts change
Policy & Security	Network wide policy application needs device level change	Policy application becomes simplified and consists

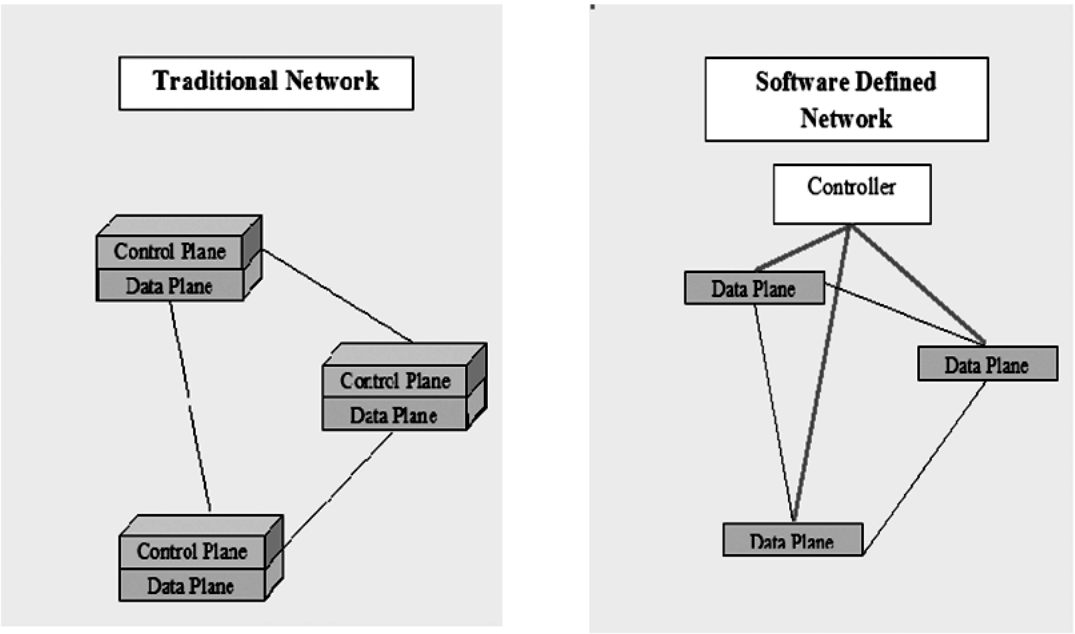

Figure 1: Difference of Traditional and Software-Defined Networks

Software-Defined Network [13] is provided the physical separation between the network control plane and data or forwarding plane. They also control the several devices. Software-Defined Network provides changing in design, build and operate network to achieve effective business activity.

Software-Defined Network is approach to computer network which are give permission administrators of network to maintain the network service through abstraction of low-level functionality.

Software is a structure propose to be dynamic manageable, cost effective and adaptable to be satisfactory for the high bandwidth, dynamic nature of present application. Programmable and virtualized for the IT service.

Software-Defined Network focuses on the three key:

- Separation of the control plane from the data plane.
- A centralized controller and global view of the network.
- Programmability of the network by outer application.

1.1. Control Plane

- Logic for control forwarding behavior.
- Make decision about where traffic is sent.

- It is signaling of the network.
- Control plane packet are destined to or locally originated by the router itself.
- It handles to software side of network.
- Routing protocol, network middle box configuration is example of control plane.

1.2. Data Plane

- Forward traffic according to control plane logic.
- Data plane know as forwarding plane.
- Data plane handle to hardware side of network.
- It deal with router, switches etc.
- IP forwarding is example of data plane.

2. ARCHITECTURE OF SDNS

Software-Defined Network architecture is plane based or layer based. In fig.2 Software-Defined Network architecture describes.

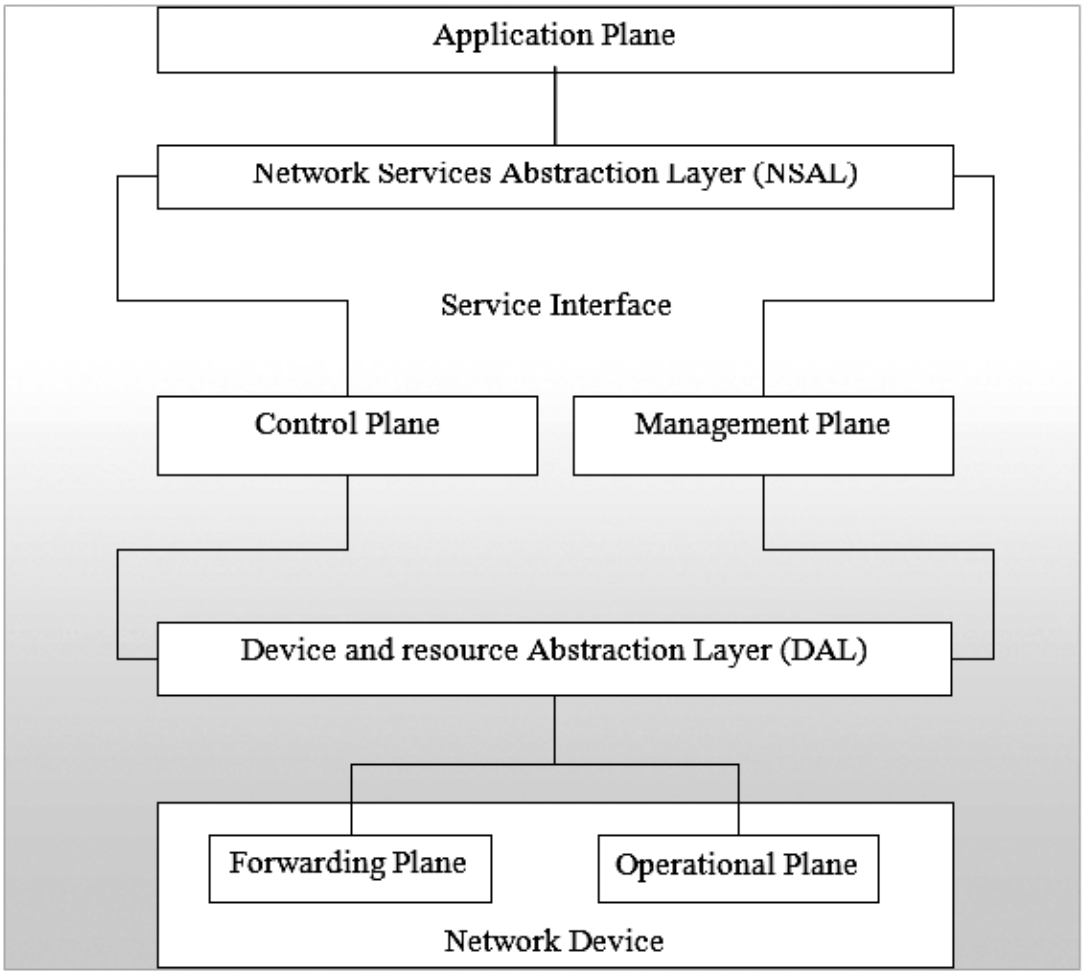

Figure 2: Software-Defined Network

3. CONGESTION

3.1. Causes

- Congestion occur when load on network is greater than the capacity of the network.
- Insufficient memory to store the packet.
- Processor work slow.

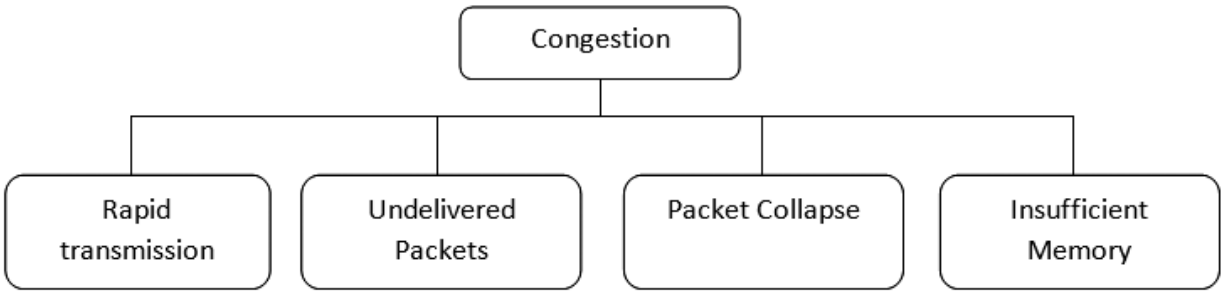

Figure 3: Causes of Congestion

4. CONGESTION CONTROL

- It is divided into two categories.
  - Congestion control is a technique to prevent congestion before happening.
  - It is also reducing the congestion after happen.

4.1. Open Loop

- Used to congestion prevention before it happen in network.
- It is deal by either sender and receiver both.

4.1.1. Retransmission Policy

The sender resend the packet if it feel that the packet loss or corrupted in way of transmission.

4.1.2. Window Policy

To implementation Window policy selective reject window method is used for congestion control.

4.1.3. Acknowledgement Policy

If the receiver does not acknowledge every packet it receive it may slow down the sender and help prevent congestion.

4.1.4. Discarding Policy

A router may discard less useful packet when congestion in likely to happen.

4.1.5. Admission Policy

A router can deny establishing a virtual circuit connection if there is congestion in the network or if there is possibility of future congestion.

4.2. Close Loop

- It mechanisms try to remove or reduce the congestion after it happens.
- It uses some kind of feedback.

4.2.1. Back Pressure

In this policy congested node stops receiving data from the immediate upstream node or nodes.

4.2.2. Choke Packet

In choke packet policy congestion occur node send directly packet to source node. No medium required as back pressure policy.

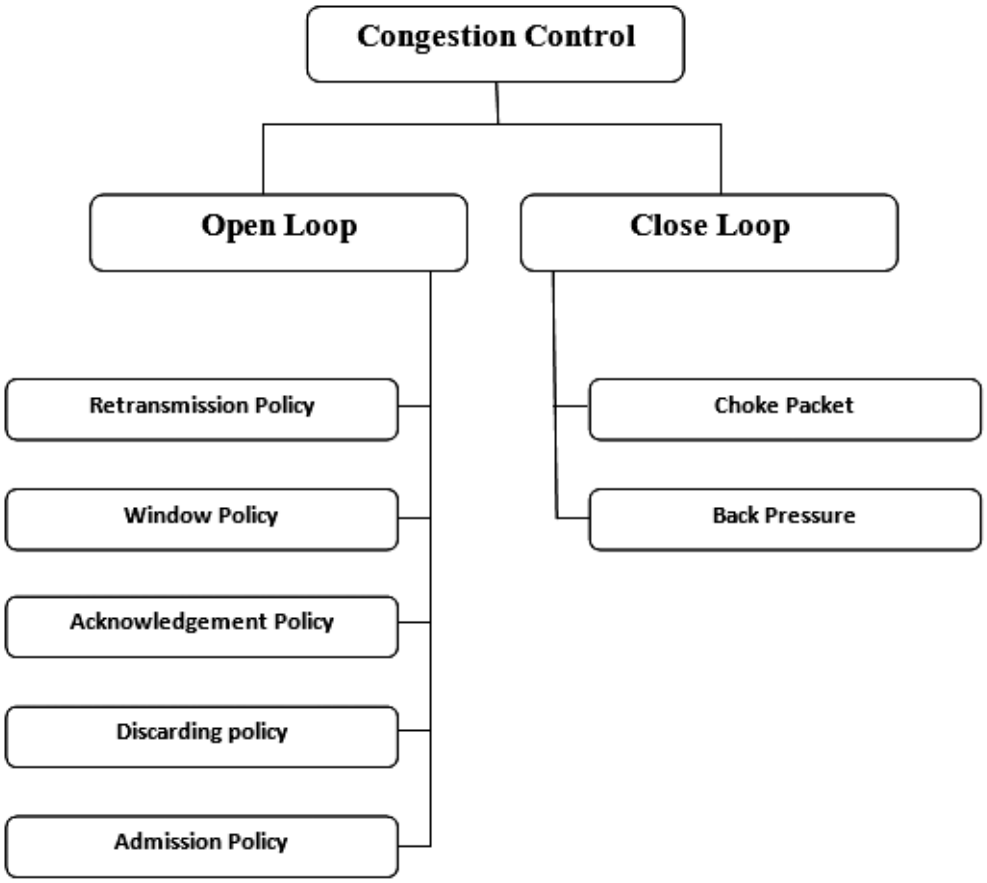

Figure 4: Classification of Congestion Control

5. RELATED WORK

According to Google Software-Defined Network is use to interconnect their data centers due to efficiency, flexibility, scalable in performing traffic engineering functions. Software-Defined Network provides new network owner and operators control of their infrastructure.

It also allowing customization, optimization, and operational costs and reducing the overall capital in network. Software-Defined Network is not mechanisms but it is a framework to solve a set of problem. It is using OpenFlow protocol. Software-Defined Network is centralization of control plane.

It should be able to manage and control thousand of device on one command. Software-Defined Network is directly programmable, centrally managed, programmatically configured, agile and open standard-based vendor neutral.

Software-Defined Network is standard southbound application program interfaces. It is used in configuration and management of network. Software-Defined Network is action in the WAN, LAN, data centre, cloud and other provider networks. Software-Defined Network is change or manage high level to low level for better communication occur in the network.

According to Narendra Kumar Jha, Namilta Agrwal, and Praveen Singh [17] describes the congestion in Software-Defined Network in Realization of congestion in Software-Defined Networks. They use to Open vSwitch [14] to found the congestion in Software-Defined Networks.

According to Yifei Lu, Shuhong Zhu [3] Software-Defined Networking is centralized method to control. When a receiver request the data from multiple sender simultaneously, occur TCP incast. In the data centre network many to one communication pattern constantly appears due to multiple servers store the data.

In SDN-based TCP congestion control in data centre network used approach active controller to select a long line flow to reduce sending rate by adjusting the TCP receiving window of acknowledgement packet after OpenFlow switch triggered a control message or acknowledgement to controller.

Masoumeh Gholami, Behzad Akbari [5] identify the performance of Data Center Network is very important for the cloud service and application, but this performance is based on different factor of network. This factor may be design of network, congestion control, security, access control etc. of data control network.

This problem solve by proposal of efficient method to solve or control the congestion in the software defined Data Center Network based on OpenFlow protocol. In this method, links are detected by centralized monitoring the OpenFlow enable switch port statistics. After that, some flow of any congestion link re-route or shift path the free resource with use OpenFlow controller.

According to Junho Suh, John Carter [6] OpenSample means a low-latency, sampling-based network measurement platform target at building faster and control loops for the Software-Defined Network. OpenSample flow the packet sampling in the network to provide real-time measurement for both individual flow and as well as network load. OpenSample is useful in any manner for network.

OpenSapmle is also particularly used in Software-Defined Network environment, where controller can take quick action based on the data OpenSample provides. This sampling is for network monitoring permission OpenSample. The sample is collect a 100 millisecond control loops. In polling-based approaches 1-5 second control loops collect.

The purpose to implement OpenSample in the Floodlight OpenFlow controller and also evaluate it both in simulation and on a test bed comprised of commodity switches. When used to inform traffic engineering, OpenFlow provides up to a maximum throughput improvement over both static equal cost multipath routing and a polling based solution with a one second control. The primary goal of John Carter is to low latency for network with flow in milliseconds. Throughput improves in some cases in network.

Masoumeh Gholami, Behzad Akbari [21] describes the issue of congestion in Data Center Network. They proposed a potential solution which is based on the OpenFlow protocol for congestion control in Data Center Networks. In this solution reroute the path of congestion link and free the resources by the OpenFlow controller.

Ina F. Akyildiz, Ahyoung Lee [7] introduce a roadmap for traffic engineering in SDN OpenFlow[12]

network, which describe Software-Defined Networking traffic engineering focuses on flow management, fault tolerance, topology update and traffic analysis of network.

Software-Defined Networking improves network resource utilization, simplify network management, efficient operation cost and promote the evolution with effective cost. Software-Defined Networking required new technique of traffic control and manage to monitor global network flow, status in traffic engineering.

In explicit multipath congestion control for data centre networks [8] based on the multipath layer in Software-Defined Networking. Yu Cao, Mingwei Xu describes the Trash algorithm to identify for shifting traffic from more congestion network to less congestion network.

Hilmi E. Egilmez, S. Tahsin Dane [9] identifies the Software-Defined Networking as a example which are separate the control and data forwarding plane of routing. Hilmi E. Egilmez proposed OpenQoS. OpenQoS is a new OpenFlow controller design for end-to-end multimedia delivery which supports Quality of Service (QoS).

In this paper approach OpenQoS based on the QoS routing where routes of multimedia traffic are optimized dynamically to fulfill the required to Quality of Service. They measure performance of OpenQoS over the real network. The performance of OpenQos compare with the performance of current state-of-the-art, HTTP based multi-bitrates adaptive streaming.

OpenQoS [9] is OpenFlow based Quality of Service. OpenQoS design for solve routing related problem in network. OpenQoS approach is different from the QoS architecture. OpenQoS provide possibility to employ dynamic routing over the network.

Hilmi E. Egilmez purposed OpenQoS for Software-Defined Network result show better performance. OpenQoS can achieve the seamless video delivery with little or no disturbances experienced by the user. If transport protocols are unreliable than used in UDP otherwise TCP. OpenQoS provide full video quality with no error.

Sugam Agrwal, Murali Kodialam and T.V.Lakshman [10] identify that Software-Defined Networking is an example of network which are separate the network into network control plane and data forwarding plane at Bell Labs. In this paper define Software-Defined Network Controller and Software-Defined Network Forwarding Element. This paper also describes the role of Software-Defined Network for traffic engineering. This paper show the use of Software-Defined Network, when Software-Defined Networks incrementally introduce into existing network.

They wants to prove that Software-Defined Network architecture provide better capacity of network and better performance in loss and delay. Software-Defined Network controller introduces solution of problem for the traffic engineering. Current papers are clarifying their solution of problem partial deployment and enhanced fast Fully Polynomial Time Approximation Schemes.

In Reliability-aware controller placement for Software-Defined Networking [11] introduce by Yannan Hu, Wang Wendong. Separated plane may be lead severe packet loss and degradation performance in the network.

Their approach and analysis for reliability of Software-Defined Networking control network is successful. They improve reliability without any latency in network.

According to Nate Foster, N. Praveen Katta [4] (2013) Language for Software-Defined Networking describe the language of Software-Defined Networking. Frenetic is a programming language introduce by Language for Software-Defined Networking.

Overall research related to Software-Defined Networking is successful in their approach and useful for us. They work for congestion control, traffic monitoring, reliability etc. in Software-Defined Networking. Table.2 describes the pros and cons of researches which are based on Software-Defined Networks.

Table 2
Study of Well Know existing protocol

No	*Author & year*	*Research Topic*	*Technique Used*	*Pros*	*Cons*
1.	N. Kumar Jha, Praveen Singh, Namita Agrwal (2015)	Realization of congest-ion in Software-Defined Network	Open vSwitch[14], iperf [15]	To find the congestion occurrence in SDNs	This only describe presence of congestion in SDNs
2.	Yifei Lu, Shuhong Zhu (2015)	SDN-based TCP congestion control in Data Center Networks	OpenTCP , SDTCP congestion control mechanism	Almost zero packet loss TCP incast	This protocol suffer from limited deployability
3.	Seungbeom Song, Kyuho Son(2016)	A congestion avoidance algorithm in SDN environment	OpenFlow configuration [12]	Improve resilience of TCP with near lossless	Only 11% throughput enhancement
4.	Junhon Suh, Colin Dixon, John Carter (2014)	OpenSample: A low latency, sample based measurement platform for commodity SDNs	OpenSample [16]	Low latency	Limited scope
5.	Yu Cao, Mingwei Xu (2013)	Explicit multipath congestion control for Data Center Network	Trash Algorithm used for shifting traffic	Centralized and demand based	Only for multipath
6.	Ina F. Akyildiz, Ahyoung Lee (2013)	A roadmap for traffic engineering in SDN OpenFlow Network	Traffic Engineering, OpenFlow [12]	Fault Tolerance, flow management	Only for limited network area
7.	Hilmi E. Egilmez, S. Tahsin Dane, A. Murat Tekalp (2013)	OpenQoS:A OpenFlow controller design for multimedia delivery with end-to-end QoS over SDNs	OpenQoS [9]	Seamless video delivery	Little noise occur in video delivery
8.	Yannan Hu, Wang Wendong, Cheng Shiduan (2013)	Reliability-aware controller placement for Software-Defined Networks	Placement controller algorithm	The problem of placing controllers in SDNs to max the reliability of control networks	Placing too many or too few controller reduce the reliability
9.	Jaehyun Hwang, Hyun-Wook Jin (2015)	Scalable congestion control protocol based on SDN in Data Center Network	Open vSwitch[14], ns-3 simulator	Minimize the latencies of service and improve the QoS in DCNs	This solution verify for under 110 worker not more than 110 worker
10.	Nate Foster, Arjun Guhn, Michael J. Freedman et al. (2013)	Language for Software -Defined Networks	OpenFlow [12]	Describe used language for SDNs	Only language description

6. PROBLEM IDENTIFICATION

In present scenario various Software-Defined Networking research occurs. This research based on the different issues of Software-Defined Networking. Issues in Software-Defined Networking are Security, access control, error minimization, connectivity, congestion control, traffic monitoring etc. Here we focus on the congestion control in Software-Defined Networking. Congestion occurs in Software-Defined Networking because of more than one packet or message receiver receives at same time. In this situation congestion occur in Software-Defined Networking.

A congestion avoidance algorithm in SDN environment [21] approach

- Congestion occurs in network if max utilization of port 70%.
- Congestion present in network, change the route of source to destination port.

Congestion occur exceed the capacity of network from the existing capacity. Traffic are increase day by day in network. It also effected to Software-Defined Network.

- How to detect the congestion in Software-Defined Network?
- How to control the congestion in Software-Defined Network?
- How to extent the range of utilization from 70%?
- Does any Congestion control algorithm existing at route discovery level?
- How to represent the Congested port in network topology?

7. RESEARCH OBJECTIVE

There are many researcher work on congestion control in Software-Defined Network. They propose various approach to avoid and detect the congestion in Software-Defined Network.

- Propose algorithm for congestion control in Software Defined Network
  - Detect the congestion
  - Reduce the congestion
  - Mark congested port
- To evaluate the performance of efficient in Software Defined Network during the congestion.
  - Calculate utilization

    Utilization % = (data bits x 100) / (bandwidth x interval)
  - Calculate hop count

    Hop count is the number of routers (number of hops) from the source router through which data must pass to reach the destination network.
- Implementation of propose algorithm using simulators.
  - Mininet
- Analysis and comparison propose and existing algorithm.

9. RESEARCH METHODOLOGY

In this section, we discuss about the method and way of approach to solve the problem. In Fig.5 we present only idea to improve the performance and method to solve the problem. To control the congestion we use the following methodology.

Congestion Control

1. Low Latency
2. Low Packet Loss
3. High throughput

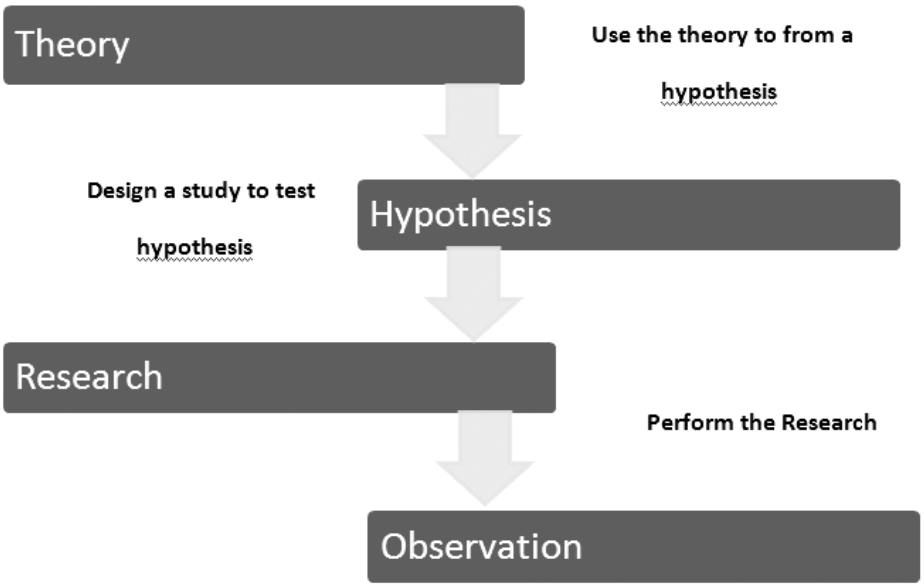

Figure 5: Methodology Of Research

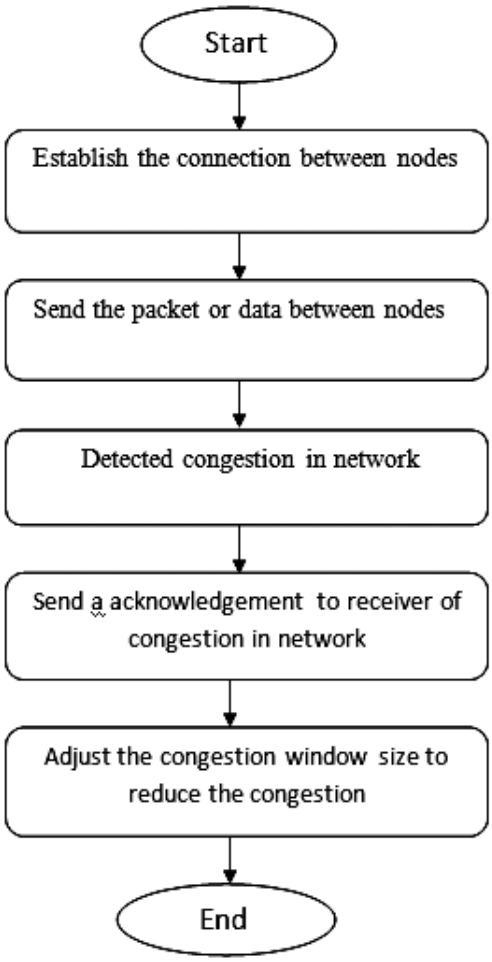

Figure 6: Flow Chart of Congestion Control for Software-Defined Network

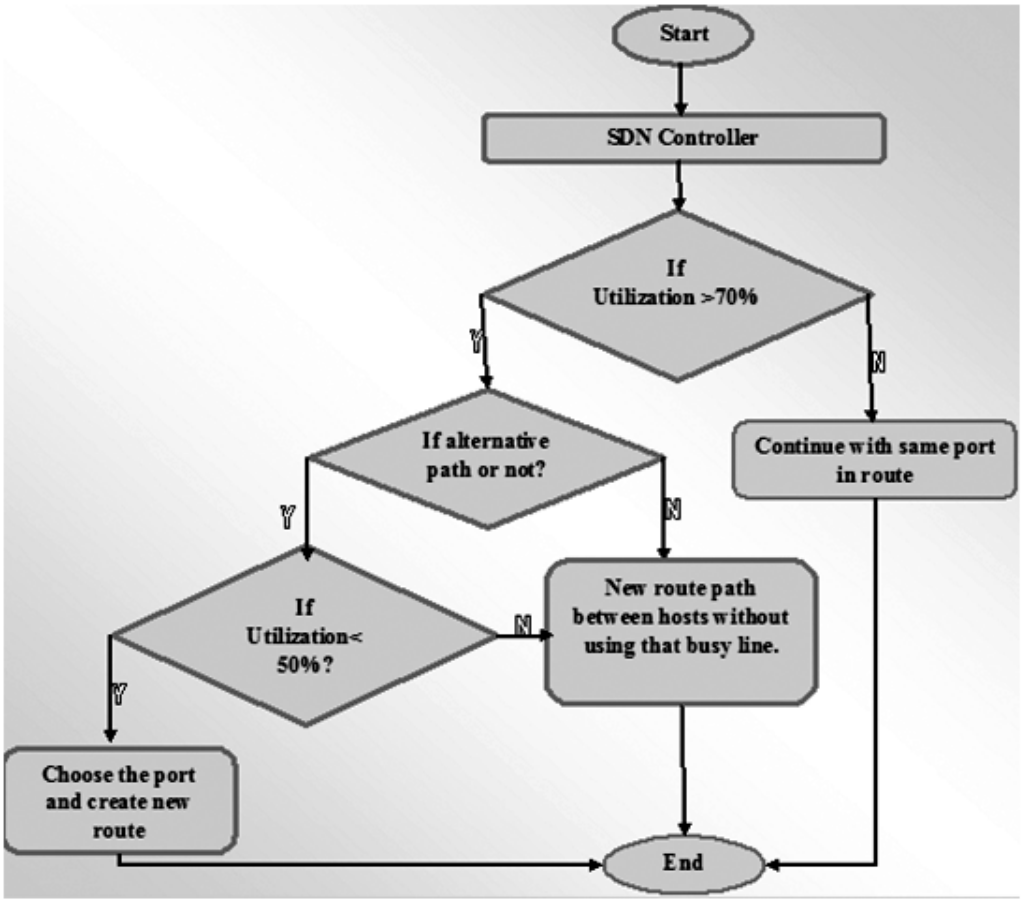

Figure 7: Flow Chart For To Detect The Congestion in Software-Defined Network

10. CONCLUSION

Our proposed method detects the congestion in Software-Defined Network. This method also reduce the congestion in the Software-Defined Network. Our methodology give a idea how to method work for congestion control in Software-Defined Network. We work on currently proposed method for better result.

REFERENCE

[1] Monia Ghobadi, Soheil Hassas Yeganeh, Yashar Ganjali, "Rethinking End-to-End Congestion Control In Software-Defined Networking" ACM, 2012

[2] David Meyer, " The Software-Defined Networking Research Group", IEEE Internet Computing, 2013

[3] Yifei Lu, Shuhong Zhu, "SDN-based TCP congestion control in data centre network", IEEE, 2016

[4] Nate Foster, Naga Praveen Katta, "Language for Software-Defined Networks", IEEE Communication Magazine, 2013

[5] Masoumeh Gholami, Behzad Akbari, " The Performance of Data Center Network", IEEE, 2015

[6] Junho Suh,Colin Dixon,Wes FilterJohn Carter,Ted Taekyoung Kwon, "OpenSanple:A low-latency sampling-based measurement platform for commodity SDN", IEEE 34th International Conference, July 2014

[7] Ina F. Akyildiz, Ahyoung Lee, Wu Chou, Min Luo, Pu Wang, "A roadmap for traffic engineering in SDN OpenFlow network", IEEE , June 2014

[8] Yu Cao, Mingwei Xu, Xiaoming Fu, "Explicit multipath congestion control for data center network", ACM 9th conference on Emerging networking experiments and technologies, December 2013

[9] Hilmi E. Egilmez, S. Tahsin Dane, "OpenQoS:An OpenFlow Controller Design for Multimedia Delivery with End-to-End Quality of Service over Software-Defined Networkings", IEEE, 2013

[10] Sugam Agrwal, Murali Kodialam, T.V. Laxman, "Traffic engineering in Software-Defined Network", IEEE INFOCOM , July 2013

[11] Yannan Hu,Wang Wendong, "Reliability-aware controller placement for Software-Defined Networking", IEEE, 2013

[12] OpenFlow.[Online]Available:http://www.opennetworking.org/home/SDN Resources/OpenFlow

[13] SDN.[Online]Available:http://www.opennetworking.org/home/SDNResources/SDN Defined

[14] Open vSwitch.[Online] Available: http://openvswitch.org

[15] iPerf. [Online] Available: https://ipref.fr

[16] Floodlight. [Online]. Available: http://floodlight.openflowhub.org

[17] Narender Kumar Jha, Namita Agrwal, Praveen Singh, "Realization of congestion in Software-Defined Networks", Computing, Communication & Automatic (ICCCA), International Conference

[18] Open Networking Foundation (ONF), "Software-Defined Networking: the new norm for networks",[Online].Available:https://www.opennetworking.org/images/stories/download/openflow/wpsdn-newnorm.pdf

[19] Open Network Foundation. [Online].Available: http://opennetworking.org

[20] S. Denazis, J. Hadi Salim, E.Halepalidi, "Software-Defined Network:Layer and Architecture Terminology", RFC 7426, Internet Research Task Force(IRTF), 2015

[21] Masoumeh Gholami, Behzad Akbari, "Congestion Control Using OpenFlow In Software Defined Data Center Networks", 19th International INIC Conference in Clouds, Internet and Networks, March 2016

[22] Seungbeom Song, Jaiyong Lee, Kyuho Son, Jihoon Lee, Hangyong Jung, "A congestion avoidance algorithm in SDN environment", IEEE Information Networking (ICOIN), 2016 International Conference on, March 2016.