A colorful logo with a black background

Description automatically generated

SUB-Computer Networks

(COM-502)2023

Faculty-DR. Navin M Upadhyay

Submission Date:30-12-23

**Implementing QOS Mechanisms in NS2 and evaluating their impact on the Performance of Wireless Networks**

**Vivek Koul (Roll No: 2021A1R080) (Leader)** \*\*\*\*\*

**Jeevanjot Singh (Roll No: 2021A1R061)** \*\*\*\*\*

**Atul Kumar (Roll No: 2021A1R072)** \*\*\*\*

**Tushant Karihaloo (Roll No: 2021A1R076)** \*\*\*\*

**Jatin Koul (Roll No: 2021A1R120)** \*\*\*\*

**Madhav Kumar Sharma (Roll No: 2021A1R075)** \*\*\*

# (GROUP-6)

Department of Computer Science & Engineering,

Model Institute of Engineering & Technology, Jammu.

Email:([Vivek Koul](mailto:2021a1r080@mietjammu.in) [Jeevanjot Singh](mailto:2021a1r063@mietjammu.in),

[Atul Kumar](mailto:2021a1r072@mietjammu.in), [Tushant Karihaloo](mailto:2021a1r076@mietjammu.in), [Jatin Koul](mailto:2021a1r120@mietjammu.in), [Madhav Sharma](mailto:2021a1r075@mietjammu.in))

**TABLE OF CONTENTS**

[ACKNOWLEDGEMENT 3](#_Toc154271735)

[Abstract 3](#_Toc154271736)

[1.Introduction: 4](#_Toc154271737)

[ *Quality of Service (QoS) in Computer Networks*: 4](#_Toc154271738)

[ *Goal of project:* 4](#_Toc154271739)

[2.Background: 4](#_Toc154271740)

[ *Challenges associated with ensuring QOS:* 5](#_Toc154271741)

[ *Brief Problem:* 5](#_Toc154271742)

[3.QoS Mechanisms Implementation (Code Snippets): 5](#_Toc154271743)

[ Pseudocode: 5](#_Toc154271744)

[ Key functions and algorithms used in the QoS implementation: 6](#_Toc154271745)

[ Simulation Parameters Setup: 6](#_Toc154271746)

[ Initialization: 7](#_Toc154271747)

[ 7](#_Toc154271748)

[ Nodes Definition: 7](#_Toc154271749)

[ Generate Movement: 7](#_Toc154271750)

[ Agents and Applications Definition: 7](#_Toc154271751)

[ Termination: 7](#_Toc154271752)

[4.) Simulation Environment Setup: 7](#_Toc154271753)

[ Parameters and configurations used for simulating the wireless network environment in NS2: 8](#_Toc154271754)

[5.) Implemented techniques for QoS: 9](#_Toc154271755)

[ Ad-Hoc On-Demand Distance Vector (AODV): 11](#_Toc154271756)

[ Dynamic Source Routing (DSR): 13](#_Toc154271757)

[6.) Experimental Design: 13](#_Toc154271758)

[ Introduction to TRACE GRAPH: 15](#_Toc154271759)

[ Simulation Result: 16](#_Toc154271760)

[7.) Results and Analysis: 16](#_Toc154271761)

[ Analysis of the 10 simulations of (AODV): 17](#_Toc154271762)

[Analysis: 17](#_Toc154271763)

[ Result Table of the 10 simulations for AODV: 18](#_Toc154271764)

[ Analysis of the additional 10 simulations of (DSDV): 18](#_Toc154271765)

[ Result Table of the 10 simulations for DSDV: 19](#_Toc154271766)

[ Analysis of the additional 10 simulations of (DSR) 19](#_Toc154271767)

[ Result Table of the 10 simulations for DSR: 21](#_Toc154271768)

[Simulations Graph Result: 21](#_Toc154271769)

[No. Of Packets received: 22](#_Toc154271770)

[Delay: 23](#_Toc154271771)

[No. Of Packets Sent: 24](#_Toc154271772)

[Packet Delivery Ratio: 25](#_Toc154271773)

[Packets Dropping Ratio: 26](#_Toc154271774)

[Throughput: 27](#_Toc154271775)

[ Capturing Team Collaboration: Behind the Scenes 29](#_Toc154271776)

[REFERENCES 28](#_Toc154271777)

# ACKNOWLEDGEMENT

We express our sincere gratitude to our faculty teacher Dr. Navin Mani Upadhyay for providing us with this assignment. We would also like to express our special thanks of gratitude to our mentor Ms. Vani Malagar for their guidance in completing the project. Her expertise and dedication were instrumental in shaping the direction of our project.

As the leader of the group, I extend my heartfelt thanks to all the members of my group for their collaborative work and dedication. The collaborative efforts of my group members were crucial in completing the assignment.

Lastly, we thank our parents for their continuous support, understanding and encouragement. Their confidence in us has always been an encouragement, and we thank them for the love and support they have given throughout this assignment completion.

We would like to express our heartfelt appreciation and gratitude for the efforts, guidance, and encouragement of everybody mentioned above without whom this assignment could not have been completed. Thank you for joining us on our academic journey.

# Abstract

This project explores the implementation and evaluation of Quality of Service (QoS) mechanisms in NS2 for wireless networks. The objective is to implement QoS mechanisms in computer networks, address challenges specific to wireless environments, and assess the impact of implemented mechanisms on network performance. We have implemented QOS to reduce the packet drop and latency of the wireless network, hence improving its efficiency. This project focuses on implementation and evaluation. Code snippets and pseudocode illustrate the implementation of key QOS mechanisms, emphasizing functions and algorithms critical to the simulation to evaluate the influence of QOS on wireless network performance measuring variables such as throughput, latency, and packet loss etc. Results are presented using visual aids, such as graphs, Nam simulator illustrating the performance under different QOS mechanisms. We are doing all of this in Ubuntu OS on which we have installed NS2, a NAM simulator. All the topological simulations run on the NAM simulator. We have implemented QOS mechanisms to increase the quality and efficiency of our wireless network. This comprehensive report contributes to the understanding of QOS in wireless networks, offering valuable insights for future improvements and implementations. Proper citations and references are included, ensuring the credibility and reliability of the information presented.

# Introduction:

## *Quality of Service (QoS) in Computer Networks*:

Quality of service (QoS) is a set of mechanisms or technology that ensures the performance of critical applications with limited network capacity. It enables organizations to adjust their overall network traffic by prioritizing specific high-performance applications. It is all about smartly arranging network resources. It makes sure important apps get the re­quired bandwidth and work quickly.

## *Goal of project:*

The goal of the project is to implement various QoS mechanisms/technologies in NS2 focusing on wireless networks. The project aims to increase the efficiency of wireless networks using different QoS techniques. By doing different experiments and simulations on multiple codes, the project seems to assess the effect of QoS on different wireless networks. The goal of the project is to make wireless communication services better and more dependable by adding Quality of Service (QoS) techniques into the Network Simulator 2 (NS2) framework. This research hopes to offer a deeper grasp on how to make network performance in changing and varied wireless environments better by offering unique insights into QOS mechanisms. This is achieved by using real world network scenarios and crucial performance indicators. Also, this project aims to contribute thorough understanding of network performance optimization in dynamic and diverse wireless settings by offering insights into ideal design of QOS mechanisms.

# Background:

* *Concept of QoS*:

The idea behind Quality of Service (QoS) in computer networks is to maximize and guarantee network service performance and dependability. It entails controlling important performance indicators including packet loss, jitter, delay, and throughput. QoS mechanisms include packet scheduling, bandwidth allocation and reserve, congestion control, and prioritization of important traffic.

## *Challenges associated with ensuring QOS:*

There are various problems that a wireless simulation may face while ensuring QoS like packet loss, high latency, etc. This packet loss and latency reduces the capability of the wireless network by. Some of the challenges include limited bandwidth, signal fluctuations, mobility disruptions, safety fears, device resource constraints, dynamic topologies, multimedia demands, quality degradation with distance, contention issues, and protocol diversity that need to be overcome to guarantee Quality of Service (QoS) in wireless networks. To maximize utilization of resources as well as enhance wireless communication reliability, these issues require adaptive mechanisms and creative solutions. Quality of Service (QoS) mechanisms are crucial in mitigating issues like congestion and latency. This project explores Quality of Service (QoS) implementation's impact on wireless networks, ensuring optimized resource allocation for enhanced performance and user satisfaction.

## *Brief Problem:*

In this paper we will analyze multiple routing algorithms to determine which routing protocol provides the best Quality of Service (QoS) keeping in view the

packet loss, latency, and delay that the resultant report will produce.

# 3. QoS Mechanisms Implementation (Code Snippets):

Pseudocode showcasing the implementation of QoS mechanisms in NS2:

We have implemented QOS using protocols like ADVOC, DSDV, DSR to improve the Quality of Service (QoS) of the wireless network as these protocols impact the QoS by influencing factors like latency, packet loss and bandwidth utilization.

## Pseudocode:

**# Simulation Parameters Setup**

Set simulation parameters for channel type,

propagation model,

network interface type,

MAC type, queue type,

link layer type,

antenna model,

maximum packet in interface queue,

number of nodes, routing protocol,

topography dimensions,

(Routing Protocol) (Protocol) #we have used three different protocols to measure the

QoS that are (DSR, AODV, DSDV)

and simulation end time.

Adjust interface queue type based on the routing protocol (AODV, DSDV, DSR).

# Initialization

Create NS simulator.

Set up topography object.

Open trace files for simulation output.

Create a wireless channel.

# Mobile Node Parameter Setup

Configure mobile node parameters: routing, MAC,

queue,

antenna,

propagation,

agent and router traces.

# Nodes Definition

Create 10 nodes with specified positions in the simulation environment.

# Generate Movement

Define movement patterns for specific nodes at certain times.

# Agents and Applications Definition

Set up agents (UDP) and connections between nodes.

Define applications (CBR - Constant Bit Rate) over connections with specific packet sizes, rates, and start/stop times.

# Termination

Define procedure for finishing the simulation.

Flush traces and close files.

Run the NAM visualization tool.

Halt the simulation.

## Key functions and algorithms used in the QoS implementation:

### Simulation Parameters Setup:

Channel & Propagation Model:

Selection of wireless channel type (Channel/Wireless Channel) and propagation model (Propagation/TwoRayGround) for signal propagation simulation.

Network Configuration:

Configuration of network interface type, MAC type, queue type, link layer type, antenna model, maximum packet limit in interface queue, number of nodes, routing protocol, topography dimensions, and simulation end time.

### Initialization:

NS Simulator Creation:

Creation of the NS (Network Simulator) instance to simulate the wireless network environment.

Topography Object Setup:

Configuration of the topography object to define the spatial layout of the simulated network.

Trace File Initialization:

Opening trace files (DSR.tr, DSR, Nam) for storing simulation output.

### Nodes Definition:

Node Creation:

Creation of 11 nodes in the simulated environment with specified positions (X, Y coordinates) for further analysis and movement simulation.

### Generate Movement:

Movement Pattern Definition:

Defining movement patterns for specific nodes at designated times to simulate mobility within the network environment.

### Agents and Applications Definition:

Agent Setup (UDP):

Setup of UDP agents and establishing connections between nodes to simulate communication.

Application Definition (CBR - Constant Bit Rate)**:**

Definition of Constant Bit Rate applications (CBR) over established connections with specific packet sizes, rates, and start/stop times to analyze QoS parameters related to data transfer.

### Termination:

Simulation Completion Procedure:

Definition of a procedure for finishing the simulation, flushing traces, closing files, visualizing the simulation using NAM, and halting the simulation.

# 4.) Simulation Environment Setup:

For simulation of wireless network in NS2(Network Simulator 2) involves the configuring of various parameters and configurations which shows the characteristics and behavior of wireless network communication.

**What is NS2?**

NS2 is a free­ tool for network experime­nts. NS2, also known as Network Simulator 2, comes from a mix of C++ and OTcl languages. It's a good standing platform. It is an open-source platform that is used by many people around the globe. NS2 is a discrete event network simulator primarily used for simulating computer networks. It is used for analyzing the behavior of multiple networks.

## Parameters and configurations used for simulating the wireless network environment in NS2:

**Node Configuration and Parameters:**

Simulator Initialization (set ns [new Simulator]): This is used to create an instance of the NS2 simulator, which is used for setting up and running the simulation.

Node Creation (set node($i) [$ns node]): In a network, nodes stand in for entities. By using this command, a new node is created and given an ID.

* **Wireless Channel Configuration:**

Channel Creation (set channel [new Channel/Wireless Channel]):The channel in wireless networks stands for the communication channel used by nodes (wired, wireless).

Propagation Model (set propagation Model [new Propagation/TwoRayGround])**:** Defines the process by which signals travel via a wireless medium.

Channel-Propagation Association ($ns at 0.0 "$channel set Propagation [$propagation Model]")**:** Connects the selected propagation model to the channel.

* **Antenna Configuration:**

Node-Antenna Association (set ant [$ns node $i]):Connects an antenna with a specific node.

Antenna Type Setting ($ant set antenna [new Antenna/Omni Antenna])**:** Sets the type of antenna, such as Omni-directional, which broadcasts signals uniform in all directions.

* **Mobility Model:**

Mobile Node Creation (set model [new Mobile Node])**:** Creates a mobile node, which represents a node with mobility.

Initial Coordinates Setting ($model set X\_ [$ns random 500]):Randomly sets the initial X-coordinate within the simulation area, simulating node movement.

* **MAC Layer Configuration:**

MAC Layer Creation (set mac [new Mac/802\_11]):Defines the Medium Access Control (MAC) layer for IEEE 802.11.

Interface Association ($mac set interface [new Phy/WirelessPhy]):Associates the MAC layer with the wireless physical layer.

* **Network Interface Card (NIC) Configuration:**

NIC Creation (set nic [new Phy/WirelessPhy/802\_11])**:** Creates a Network Interface Card (NIC) with the given wireless technology.

Bandwidth Settings ($nic set bandwidth\_ 2Mb):Sets the bandwidth of the NIC, influencing the data rate of the wireless link.

* **Traffic Generation:**

Traffic Source Creation (set tcp [new Agent/TCP] or set cbr [new Application/Traffic/CBR]):Defines the type of traffic source, such as TCP or CBR.

* **Routing Protocol Configuration:**

Routing Protocol Selection (set rp [new Agent/rp)**:** Chooses a routing protocol for the simulation.

Protocol-Specific Parameter Configuration ($rp set something value):Sets parameters specific to the chosen routing protocol (AODV, DSR, DSDV).

* **Simulation Time and Events:**

Simulation Time Setting ($ns at <time> "$ns halt"):Sets the simulation time to a specific value and halts the simulation at that time.

* **Visualization (Optional):**

Trace and NAM File Configuration (set tracefd [open out.tr w] and $ns namtrace-all [open out.nam w] $tracefd):Optional step for visualization purposes. Trace files record events, while NAM files can be used with the Network Animator tool for visualizing the simulation.

# 5.) Implemented techniques for QoS:

For implementing Quality of Service (QoS) in wireless simulation different methods can be used. For our assignment we have used multiple routing protocols.

* **ROUTING PROTOCOLS:**

Whenever a packet wants to be transmitted via different nodes in the network, a routing protocol is required. All such protocols find a suitable route to deliver the packets to the proper destination.

Routing protocols: A set of rules and conventions governing how routers determine the best route-that is, path or link-for data transmission from source node (or source) to destination within computer networks. Routers use these to exchange routing information, and from this they build tables containing that information.

The different classifications of routing protocols are:

**Proactive, Reactive and Hybrid protocols.**

* **Proactive protocol**: In these types of routing protocol, every node retains routing information about the other nodes within the network via the routing tables, which are changed regularly as the topology of the network changes. The diﬀerent types of proactive protocols are:

Destination Sequenced Distance Vector (DSDV), Wireless Routing Protocol (WRP), Fisheye State Routing Protocol (FSR), Optimized Link State Routing Protocol (OLSR), Cluster Gateway Switch Routing protocol (CGSR), Topology Dissemination Based on Reverse Path Forwarding (TBRPF), etc.

* ***Reactive protocol*:** The routes are generated when they are required. While the destination is reachable or until the route is no longer required, the route remains valid. This kind of routing protocol is best suited for FANETs but this form of routing results in high latency and there is no routing protection. The diﬀerent types of Reactive protocols are:

Ad-Hoc ON Demand Distance Vector (AODV), Dynamic Source Routing Protocol (DSR), Temporally Ordered Routing Algorithm (TORA), Associativity Based Routing (ABR), etc.

* ***Hybrid protocol*:** To overcome the proactive routing protocols routing overhead problems and reactive protocols high latency problem, the hybrid protocol is proposed.

The diﬀerent types of hybrid protocols are:

Hybrid Wireless Mesh Protocol (HWMP), Zone Routing protocol (ZRP), Secured Hierarchical Anonymous Routing Protocol (SHARP), Temporally Ordered Routing Algorithm (TORA), etc.

**Ad-Hoc Routing Protocol**

**Proactive Routing Protocol**

**Reactive Routing Protocol**

**Hybrid Routing Protocol**

**OSLR,DSDV,GSR etc.**

**AODV,LAR,DSR etc.**

**ZRP,ZHLS,CEDAR etc.**

**Fig1**: Classification of routing protocols

For checking the quality of service, we have implemented AODV, DSDV, DSR protocols.

## 

## Ad-Hoc On-Demand Distance Vector (AODV):

AODV a reactive routing protocol is designated as one of the finest routing protocols with outstanding flexibility to cope to rapidly changing communication demands, use low processing and lower overhead memory, low network consumption and efficient of determine unicast routes from

source node to destination node with loop avoidance.

AODV routing protocol comprises of three phases:

(i) Route discovery

(ii) Packet Transmitting

(iii) Route Maintaining.

If a source UAV wants to send a packet, it initiates a route discovery operation first to detect the location of the intended UAV and then forward the packet over a specified route without making a loop during the packet transmission process. The maintenance process of the route is performed to recover link failure.

***Fig-2:* Ad-Hoc On-Demand Distance Vector Routing Sample Network**

* + **Destination Sequenced Distance Vector (DSDV):**

The Destination-Sequenced Distance Vector (DSDV) protocol is the commonly used proactive routing protocol in mobile ad hoc networks (MANET). In DSDV, each node maintains a routing table with one route entry for each destination in which the shortest path is recorded. It uses a destination sequence number to avoid routing loops.

The routing table is maintained in this protocol which includes three parameters like Destination, Distance and Next Hop. Each node transmits the routing table at a specific interval nearby nodes and other nodes recalculate the parameters. DSDV has been designed to limit the distance vector routing restrictions and add to them two route parameters: sequence number, damping.

***Fig-3:* Destination Sequenced Distance Vector Routing Sample Network**

## Dynamic Source Routing (DSR):

The DSR protocol is a reactive, wireless multi-hop routing protocol, that is also used for the routing of data packets between UAVs in the FANETs. UAV sends the data packet with a Request ID under the following protocol to prevent congestion of any kind. Under DSR, each source node saves the path from its position in the data header to its destination, where any kind of network problem like the connection Failure is performed to create new paths.

***Fig-4*: Dynamic Source Routing Sample Network**

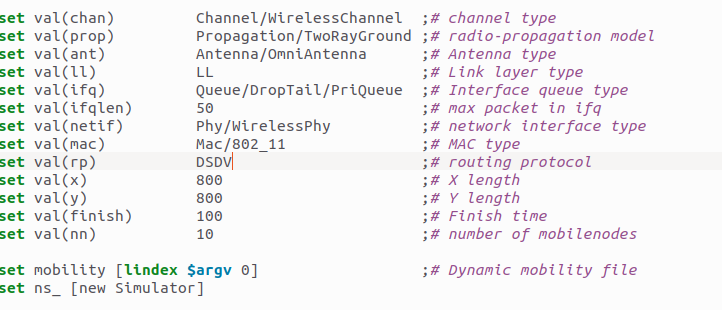
# 6.) Experimental Design:

In our experiment we had conducted various experiments among different wireless networks to see which routing protocol gives the best throughput. We experimented with multiple wireless network topologies, learnt about their simulation in the NAM and performed multiple experiments on it.

The NAM simulator emerged as a critical tool in our analysis, providing a visual representation of packet dynamics, drop rates, and node communications. The graphical insights gleaned from the simulator allowed us to discern patterns, bottlenecks, and efficiency levels associated with each routing protocol. Instances of packet drops were meticulously examined to unravel the underlying factors contributing to these occurrences, offering a comprehensive view of the protocols' robustness and resilience.

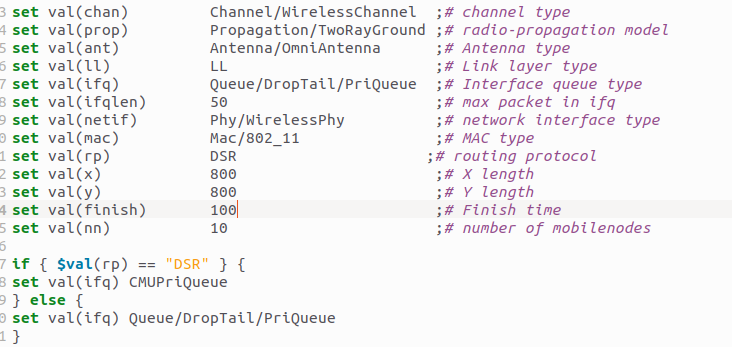
At the end we made our own wireless network that consists of 10 nodes. We saw the simulation through NAM simulator where we saw the packet drop and the communication between the nodes.

We implemented three routing protocols i.e. DSDV, DSR, AODV as shown in figures below:



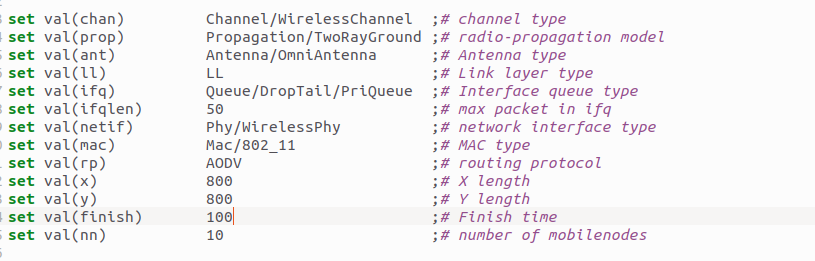
***Fig-5:* DSDV Protocol**

In figure-5 we have implemented DSDV protocol in line number 6 on a wireless simulation. So here in DSDV protocol, this protocol uses the sequence number assigned to the Destination node to resist looping and network congestion when there is any change in the topology occurs. Every UAV with a stronger sequence number is faster and more efficient than the low UAV sequencing number.



***Fig-6: DSR* Protocol**

In figure-6 we have implemented DSR protocol: To store previously found routes, it uses a source routing technique and maintains a route cache. It uses hop-by-hop routing and doesn't require a routing database to be updated. Its low overhead results from the fact that routing tables do not need to be updated continuously. Nevertheless, there may be a drawback to the increased latency brought on by having to identify a packet's entire path before delivering it.



***Fig-7: AODV* Protocol**

In Figure-7 we implemented AODV protocol: AODV uses distance vector technique. It maintains a routing table of all the nodes and their respective distances, using sequence numbers to avoid routing loops as well as ensure freshness of routes. But one advantage of this is that it allocates network resources economically, creating routes only as they are needed. However, changing the routing table is not easy--not even for smaller networks.

As we implemented these three protocols, we did the analysis of the results that we were generating with the help of TRGRAPH.

## Introduction to TRACE GRAPH:

TRACE GRAPH is a network analyzer tool which is used to provide analysis of the network that we have made be it wired or wireless. Trace Graph provides us with the analysis of things like Packet Loss, Packet Forwarding and much more.

The main function of Trace Graph is that it provides Graphs to show the graphical representations of our analysis be it throughput, delay, forwarding of packets and many more.

* + ***Performance Metrics:***

We have used different network performance metrics through our simulation between TCP and UDP. These performance metrics are used to evaluate the performance of the protocols.

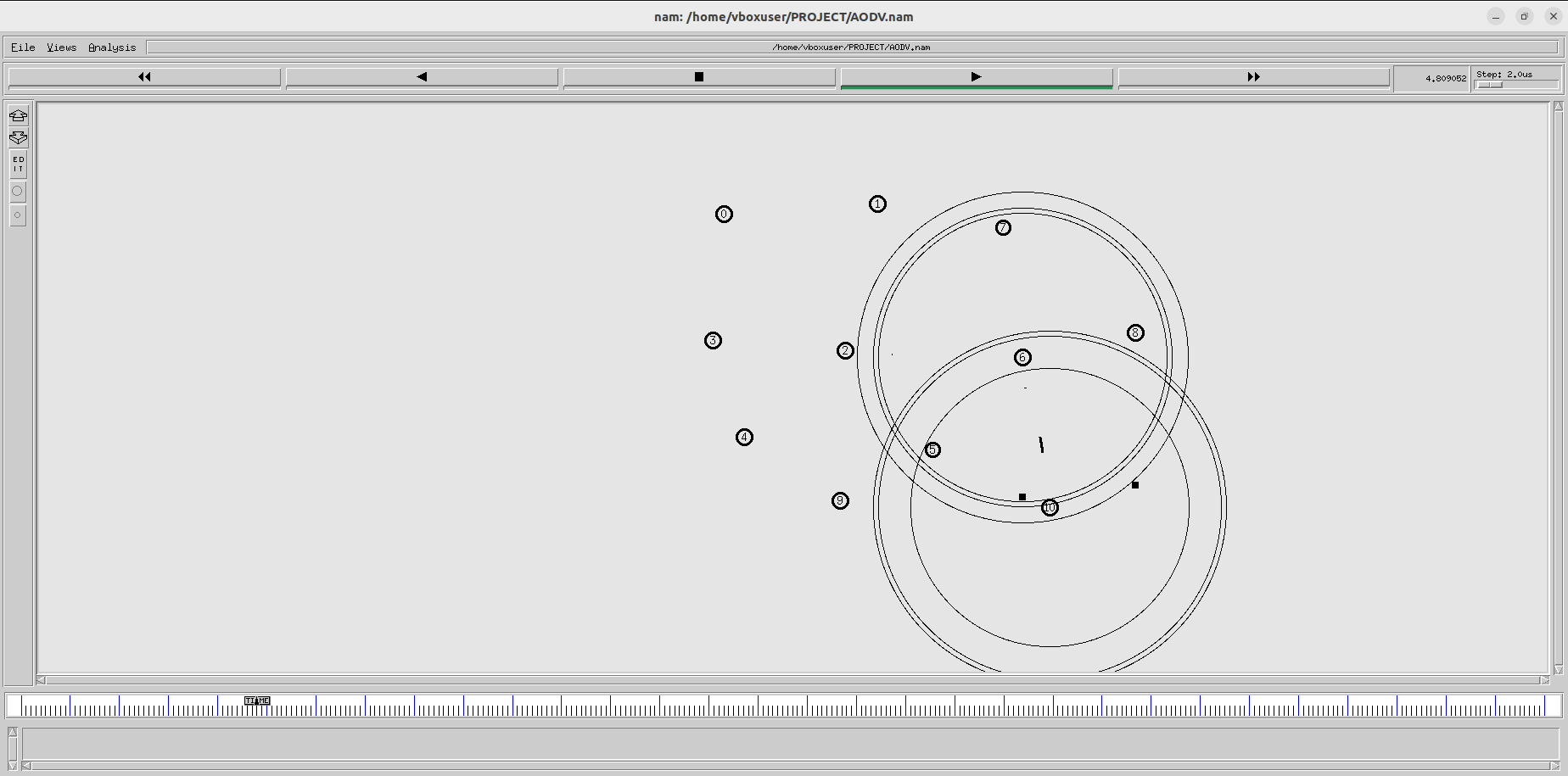
* + ***TCP Protocol:***

This is a connection-oriented protocol, one that establishes a secure and logical channel between the sender and receiver before data transmission begins. This connection setup follows a three-way handshake.

* + ***UDP Protocol***:

UDP is a connectionless protocol. That means it doesn't set up a dedicated link before sending data. This is a simpler, speedier protocol than TCP.

## Simulation Result:



***Fig-8:*** *Simulation* of network

There are 10 nodes in the simulation. This simulation shows the forwarding of packets and the packet drop the “dash "represents packet forwarding and “square” represents the packets that are being dropped.

# 7.) Results and Analysis:

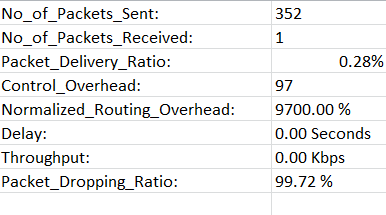
In our simulation, we have compared and evaluated the performance of three different types of routing protocols, they are DSR, AODV which are considered reactive routing protocols and DSDV which is a proactive routing protocol. The simulation has been used NS-2.35 in MANET. Ns-2.35 is software which can handle huge amounts of data efficiently. Also, it is very commonly used software, because of its high capacity to carry out complex mathematical calculations, data analysis and simulations. In this study, the simulation scenario has a different number of nodes. According to the simulation results obtained, this shows the performance results of the routing protocols in the throughput. Where DSR outperforms the DSDV and AODV, DSR has achieved 14.834 of throughput. The performance of AODV protocol is slightly lower than DSR protocol where AODV has achieved 13.764 throughput and DSDV protocol has obtained 7.822. The table below shows the results of AODV, DSDV, and DSR. This result may vary depending on the number of nodes. If the nodes are less than 20, DSR outperforms other protocols and if the nodes are more than 80 then AODV may outperform the other protocol. Condition greatly matters in the performance of protocols.

## Analysis of the 10 simulations of (AODV):

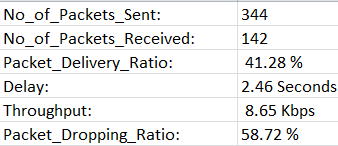
Below is the provided analysis of those additional 10 simulations using AODV protocol and further graphs are also provided.

## Analysis:

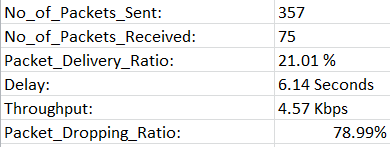
**########## Simulation number 1 ##########**



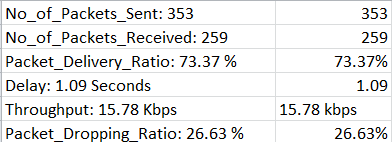
**########## Simulation number 2 ##########**



**########## Simulation number 3 ##########**



**########## Simulation number 4 ##########**



All other simulations data are provided in the table below:

## Result Table of the 10 simulations for AODV:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No. of simulation** | **Packets Sent** | **Packets**  **Received** | **Packets delivery**  **ratio** | **Delay** | **Throughput** | **Packets dropping ratio** |
| 1 | 352 | 1 | 0.28 | 0 | 0 | 99.72 |
| 2 | 344 | 142 | 41.28 | 2.46 | 8.65 | 58.72 |
| 3 | 357 | 75 | 21.01 | 6.14 | 4.57 | 78.99 |
| 4 | 353 | 259 | 73.37 | 1.09 | 15.78 | 26.63 |
| 5 | 353 | 139 | 39.38 | 2.77 | 8.46 | 60.62 |
| 6 | 354 | 353 | 99.72 | 0.3 | 21.53 | 0.28 |
| 7 | 344 | 342 | 99.42 | 0.01 | 20.84 | 0.58 |
| 8 | 353 | 351 | 99.43 | 0.06 | 21.37 | 0.57 |
| 9 | 345 | 345 | 100 | 0.01 | 20.98 | 0 |
| 10 | 349 | 254 | 72.78 | 0.01 | 15.46 | 27.22 |
| Average | 350.4 | 226.1 | 64.667 | 1.285 | 13.764 | 35.333 |

## Analysis of the additional 10 simulations of (DSDV):

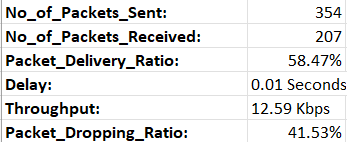
**########## Simulation number 1 ##########**



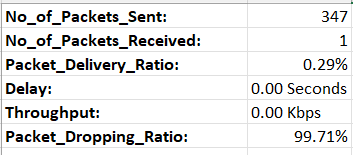
**########## Simulation number 2 ##########**



**########## Simulation number 3 ##########**



**########## Simulation number 4 ##########**



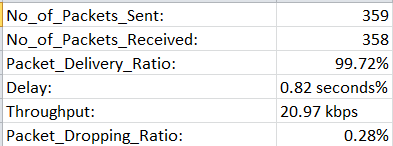
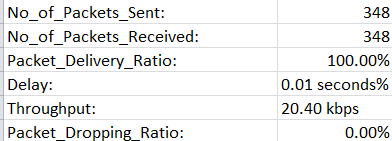
All other Simulation data are provided in the data below:

## Result Table of the 10 simulations for DSDV:

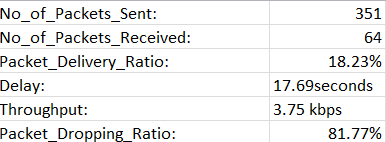
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No. of simulation** | **Packets Sent** | **Packets**  **Received** | **Packets delivery**  **ratio** | **Delay** | **Throughput** | **Packets dropping ratio** |
| 1 | 349 | 1 | 0.29 | 0 | 0 | 99.71 |
| 2 | 350 | 241 | 68.86 | 0.01 | 14.69 | 31.14 |
| 3 | 354 | 207 | 58.47 | 0.01 | 12.59 | 41.53 |
| 4 | 347 | 1 | 0.29 | 0 | 0 | 99.71 |
| 5 | 359 | 79 | 22.01 | 0.02 | 4.81 | 77.99 |
| 6 | 349 | 141 | 40.4 | 0.01 | 8.58 | 59.6 |
| 7 | 354 | 1 | 0.28 | 0 | 0 | 99.72 |
| 8 | 349 | 233 | 66.76 | 0.01 | 14.2 | 33.24 |
| 9 | 350 | 32 | 9.14 | 0.02 | 1.95 | 90.86 |
| 10 | 352 | 352 | 100 | 0.01 | 21.4 | 0 |
| Average | 351.3 | 128.8 | 36.65 | 0.009 | 7.822 | 63.35 |

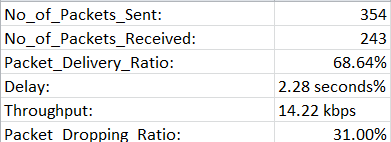
## Analysis of the additional 10 simulations of (DSR)

**########## Simulation number 1 ##########**

  
  
  
**########## Simulation number 2 ##########**  
  
 

**########### Simulation number 3 ##########**

  
  
 **########## Simulation number 4 ##########**



All other Simulation data are provided in the data below.

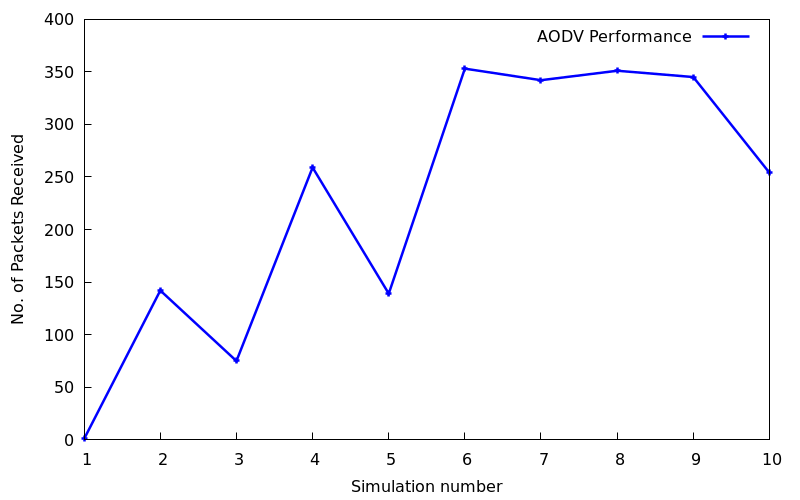
## Result Table of the 10 simulations for DSR:

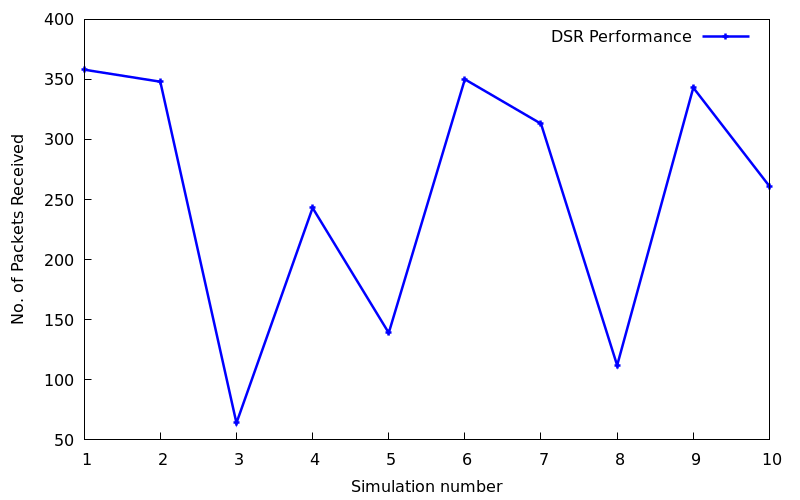
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No. of simulation** | **Packets Sent** | **Packets**  **Received** | **Packets delivery**  **ratio** | **Delay** | **Throughput** | **Packets dropping ratio** |
| 1 | 359 | 358 | 99.72 | 0.82 | 20.97 | 0.28 |
| 2 | 348 | 348 | 100 | 0.01 | 20.4 | 0 |
| 3 | 351 | 64 | 18.23 | 17.69 | 3.75 | 81.77 |
| 4 | 354 | 243 | 68.64 | 2.28 | 14.22 | 31.36 |
| 5 | 349 | 139 | 39.83 | 2.15 | 8.16 | 60.17 |
| 6 | 350 | 350 | 100 | 0.89 | 20.51 | 0 |
| 7 | 354 | 313 | 88.42 | 1.51 | 18.36 | 11.58 |
| 8 | 343 | 112 | 32.65 | 4.09 | 6.55 | 67.35 |
| 9 | 343 | 343 | 100 | 0.97 | 20.14 | 0 |
| 10 | 344 | 261 | 75.87 | 1.81 | 15.28 | 24.13 |
| Average | 349.5 | 253.1 | 72.336 | 3.222 | 14.834 | 27.664 |

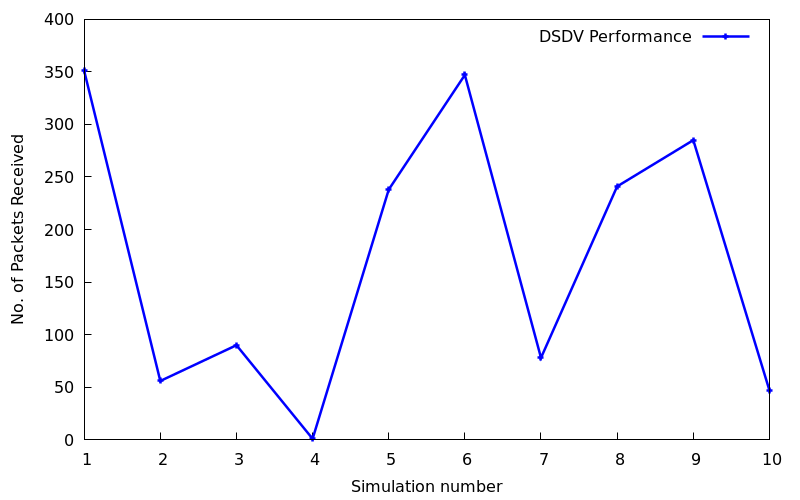
## Simulations Graph Result:

Given below are the graphs of the acquired values from the result ta

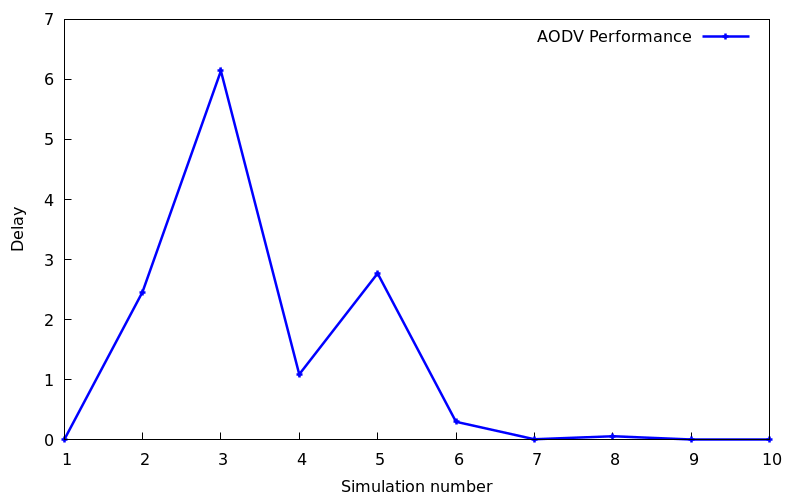
## No. Of Packets received:

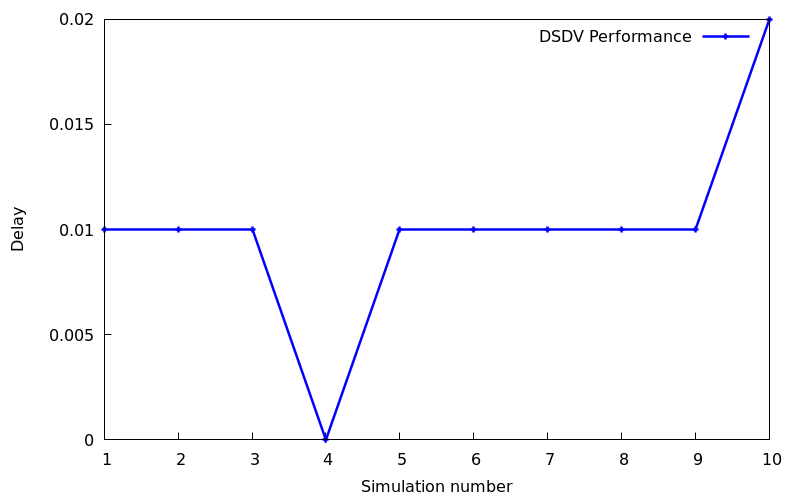


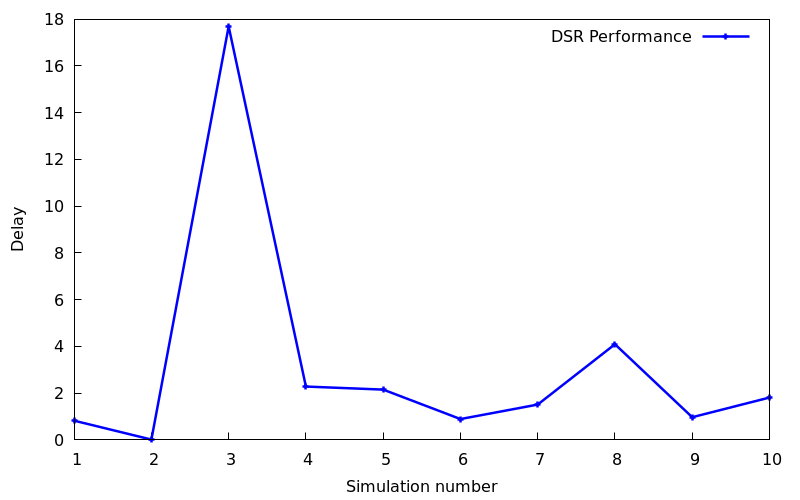




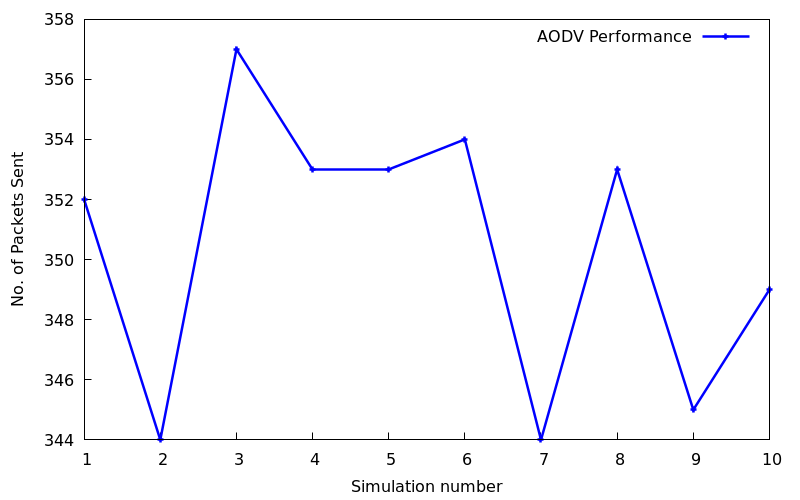
## Delay:

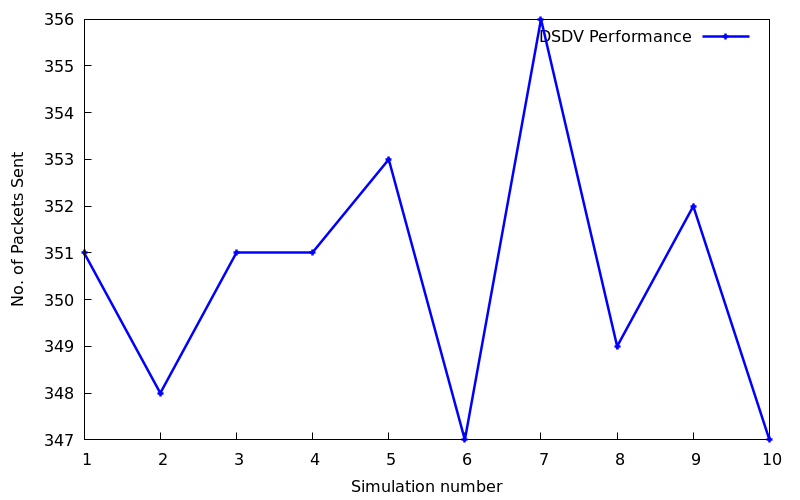


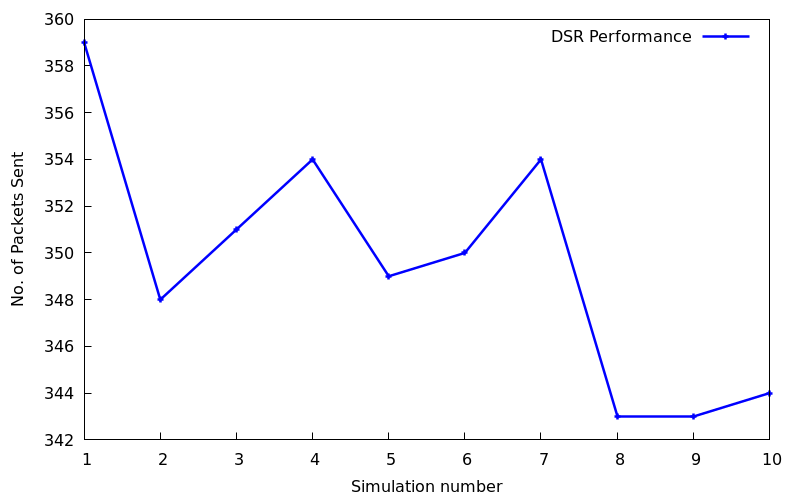




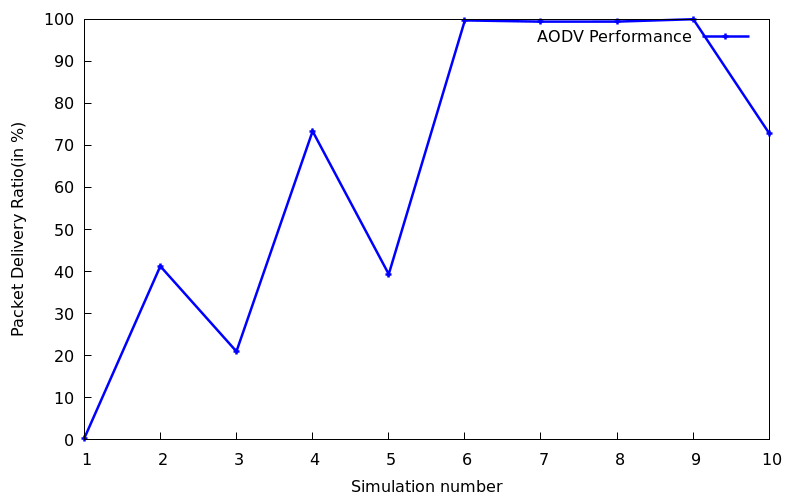
## No. Of Packets Sent:

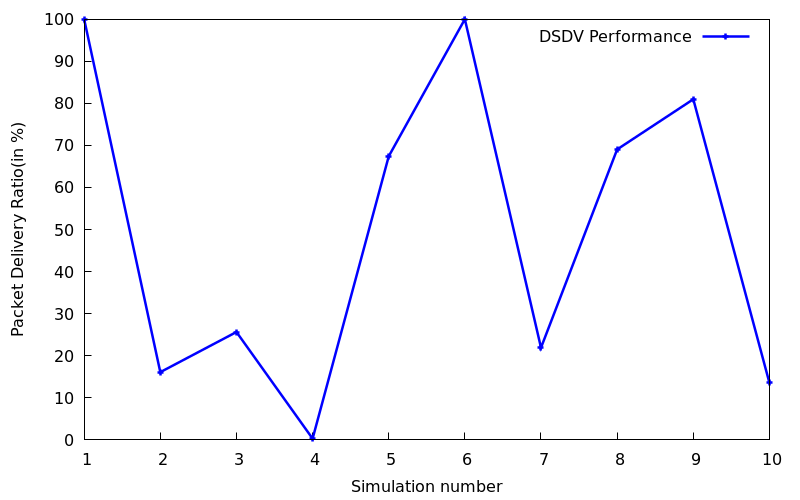


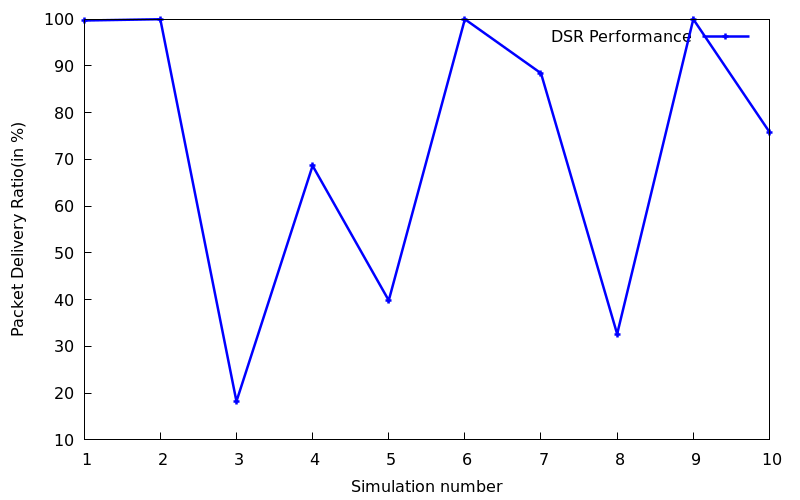




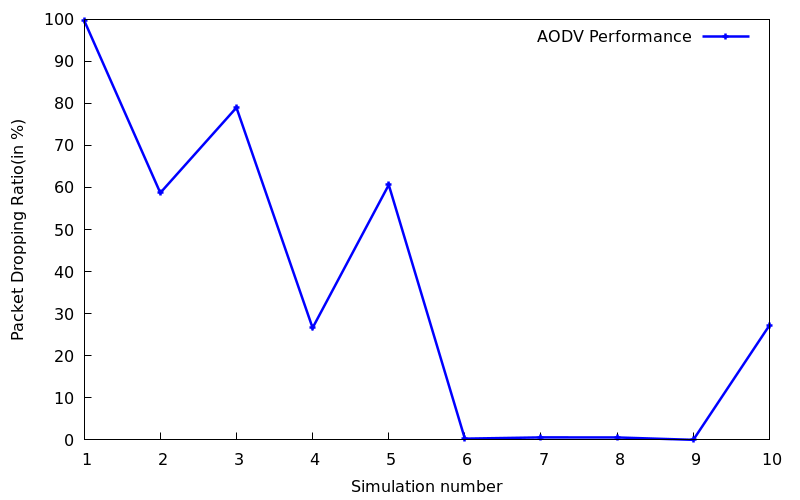
## Packet Delivery Ratio:

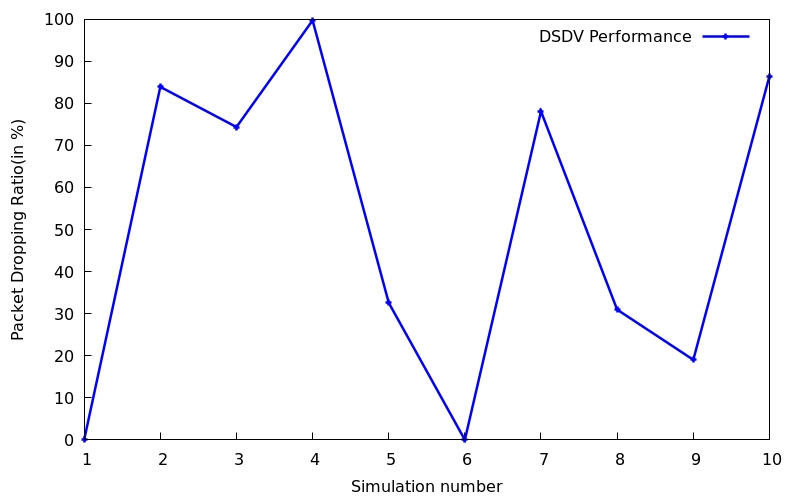


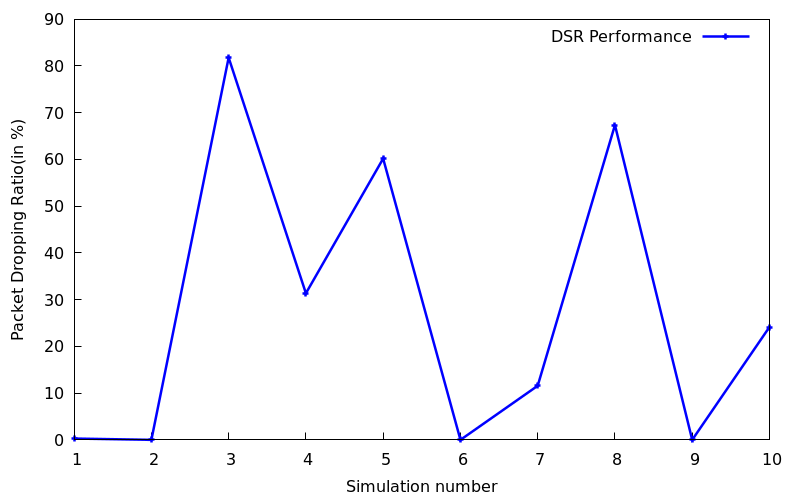




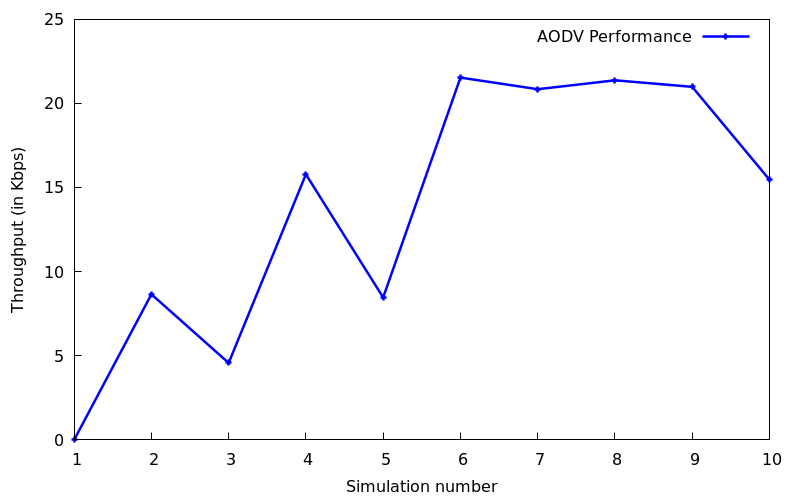
## Packets Dropping Ratio:

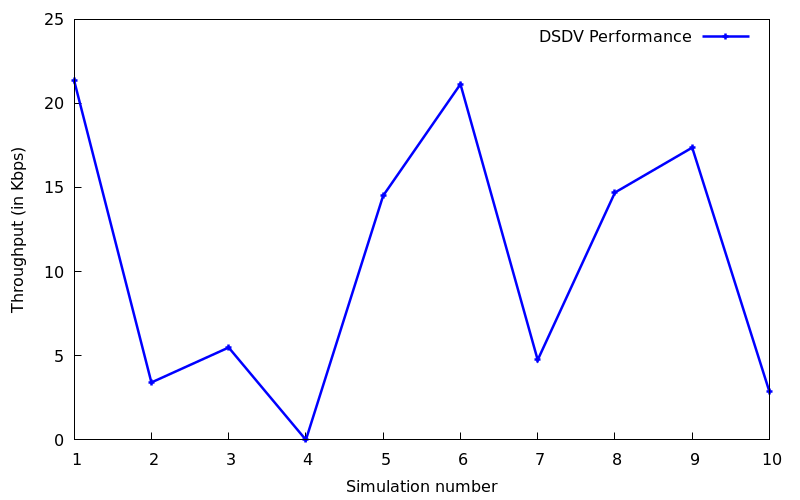


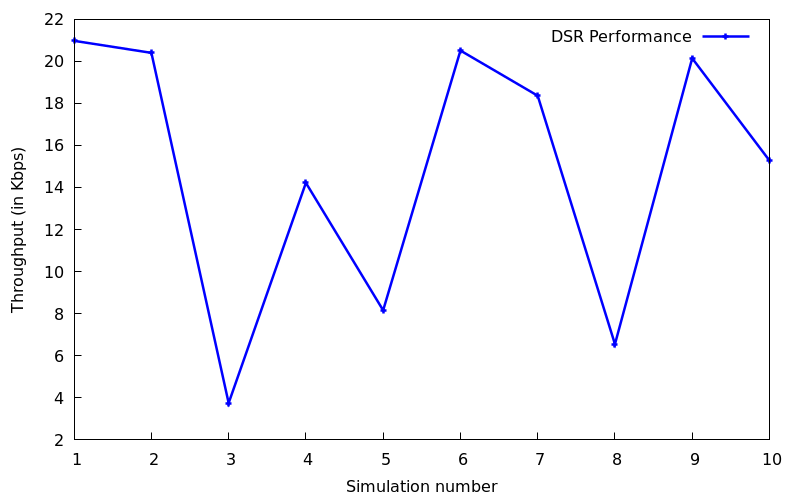




## Throughput:







8)**CONCLUSION**:

After implementing multiple routing protocol on a simulation for at least 10 times we found out that DSR protocol provides the best QoS as DSR provides best average throughput among all other routing protocols also by analyzing the graph we can easily conclude that DSR provides the best QoS and after that AODV and then DSDV.

Note this is result is based on our experiment as we had set 10 nodes, but the result may vary for a different device as there may be increase in size and number of nodes.

So, concluding DSR provides the best Quality of Service (QoS) among all other routing protocols.

# REFERENCES

# YouTube: [Engineering Clinic For installing NS2 and NAM](https://youtu.be/tH0yrJdovWM?feature=shared)

# Geek For Geeks[: Learned about protocols from there](https://www.geeksforgeeks.org/what-is-the-difference-between-aodv-and-dsr/#:~:text=It%20is%20a,in%20bigger%20networks.)

# Fortinet: [For some basic knowledge of QoS](https://www.fortinet.com/resources/cyberglossary/qos-quality-of-service#:~:text=Quality%20of%20service%20(QoS)%20is,networks%20(WANs)%2C%20and%20service%20provider)

# Research Paper: [MANET Routing Protocols Evaluation: AODV, DSR and DSDV](https://matec-conferences.org/articles/matecconf/pdf/2018/09/matecconf_mucet2018_06024.pdf)

# Other than this we also took some basic information from ChatGPT.

## Capturing Team Collaboration:

A screenshot of a computer

Description automatically generatedA screenshot of a video chat

Description automatically generated

A group of men sitting at a table with computers

Description automatically generated

**\*Stars out of 5 have been provided to group members according to their work in assignment on the first page. \***