

**AN IMPROVED SECURITY ALGORITHM FOR V2V  
COMMUNICATION USING MACHINE LEARNING**

Thesis Submitted for the Award of the Degree of  
**DOCTOR OF PHILOSOPHY**

**in**  
**Computer Applications**

**By**  
**Heena Khanna**  
**Registration Number: 41700090**

**Supervised By**  
**Dr. Manmohan Sharma**  
**Professor**  
**School of Computer Application**  
**Lovely Professional University**

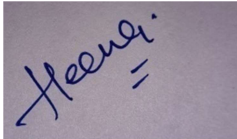


**LOVELY PROFESSIONAL UNIVERSITY, PUNJAB**  
**2023**

## DECLARATION

---

I, hereby declared that the presented work in the thesis entitled “An Improved Security Algorithm for V2V Communication using Machine Learning” in fulfilment of my degree of **Doctor of Philosophy (Ph.D.)** is an outcome of research work carried out by me under the supervision of Dr. Manmohan Sharma, working as Professor, in the Department of Computer Applications of Lovely Professional University, Punjab, India. In keeping with the general practice of reporting scientific observations, due acknowledgments have been made whenever the work described here has been based on the findings of other investigators. This work has not been submitted in part or full to any other University or Institute for the award of any degree.



Heena Khanna

41700090

Department of Computer Applications

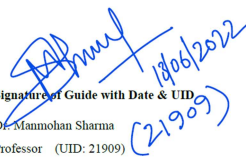
Lovely Professional University,

Punjab, India

# CERTIFICATE

---

This is to certify that the work reported in the Ph.D. thesis “An Improved Security Algorithm for V2V Communication using Machine Learning” submitted in fulfillment of the requirement for the reward of the degree of **Doctor of Philosophy (Ph.D.)** in the Department of Computer Application, Lovely Professional University, is a research work carried out by Heena Khanna, 41700090, is a bonafide record of her original work carried out under my supervision and that no part of the thesis has been submitted for any other degree, diploma or equivalent course.

  
Signature of Guide with Date & UID  
Dr. Manmohan Sharma  
Professor (UID: 21909)

Dr. Manmohan Sharma

Professor

Department of Computer Applications

Lovely Professional University

Punjab-144411 India

## ACKNOWLEDGEMENT

---

It is my immense pleasure to pay my sincere thanks to all those who have helped me to accomplish this Ph.D. thesis. Firstly, I wish to express my deepest gratitude to **Dr. Manmohan Sharma** for guiding me throughout this research work. My supervisor has been a continuous source of knowledge, inspiration, motivation, and encouragement during the entire course of this research work.

A special thanks to the management of Lovely Professional University for supporting me in the best possible manner and facilitating me in balancing my work and my research. The doctoral program of LPU has made it possible for me to pursue my dream of research and upgrading my knowledge.

I am indebted to the Examiners of end-term presentations and reviewers of journals who vetted my submissions and gave their priceless feedback to further improve the work.

I am immensely grateful to my husband **Mr. Paurush Kalra** for his never-ending support, love & belief in me without which this process would have become impossible. I owe this degree and my growth to him. I wish to express my profound gratitude to my parents and all other members of my family. Their love, support, and unshakable faith in me provide me strength to achieve all my goals in life.

My mother-in-law, **Mrs. Sunita Kalra** has been substantial throughout this period of research with her unwavering support. Her contribution to my life is beyond any acknowledgment.

I also take this opportunity to express heartfelt gratitude to all my teachers who have shaped me and have contributed enormously to my growth, knowledge, and skill development since childhood.

Finally, I would like to thank each person who has, directly and indirectly, inspired me to keep growing continuously in life.

# TABLE OF CONTENTS

---

|  |      |
|--|------|
| DECLARATION .....  | ii   |
| CERTIFICATE .....  | iii  |
| ACKNOWLEDGEMENT .....  | iv   |
| TABLE OF CONTENTS.....                                       | v    |
| LIST OF FIGURES .....  | ix   |
| LIST OF TABLES.....  | xii  |
| ABBREVIATIONS .....  | xiii |
| ABSTRACT.....  | xvi  |
| Chapter 1: Introduction.....                                 | 1    |
| 1.1    Vehicular Ad-Hoc Network (VANET).....                 | 1    |
| 1.2    Vehicular Networks (VANETs).....                      | 2    |
| 1.2.1    Concept .....                                       | 3    |
| 1.2.2    Comparison of VANET and MANET .....                 | 4    |
| 1.3    Components of VANET.....                              | 5    |
| 1.3.1    Important Component .....                           | 5    |
| 1.3.2    Types of Communication.....                         | 8    |
| 1.4    Challenges in VANETs.....                             | 10   |
| 1.5    Routing (routing protocols and their challenges)..... | 12   |
| 1.5.1    Protocols Based on Geographical Information .....   | 12   |
| 1.5.2    Broadcast Based Routing Protocols.....              | 12   |
| 1.5.3    Cluster Based Routing Protocols .....               | 13   |
| 1.5.4    Geo-cast Routing Protocols .....                    | 13   |

|                                       |   |    |
|---------------------------------------|---|----|
| 1.5.5                                 | Routing Protocols Based on the Topology .....     | 13 |
| 1.5.6                                 | Proactive Protocols .....                         | 14 |
| 1.5.7                                 | Protocols Based on the Reactive Nature .....      | 15 |
| 1.5.8                                 | Protocols Based on Hybridization .....            | 16 |
| 1.6                                   | Applications of VANETs .....                      | 17 |
| 1.6.1                                 | Traffic Application for Safe Road .....           | 17 |
| 1.6.2                                 | Applications for Informative Purposes .....       | 18 |
| 1.6.3                                 | Attacks and Threats .....                         | 19 |
| 1.7                                   | Machine Learning .....                            | 20 |
| 1.8                                   | Organization of the Thesis .....                  | 21 |
| 1.9                                   | Summary .....                                     | 21 |
| Chapter 2: Literature Review .....    |   | 22 |
| 2.1                                   | Survey of Existing Research .....                 | 22 |
| 2.2                                   | V2V Communication Based on Machine Learning ..... | 31 |
| 2.3                                   | Survey of Existing Routing Protocols .....        | 33 |
| 2.3.1                                 | Reactive (On-Demand) Protocols .....              | 34 |
| 2.3.2                                 | Proactive Protocols .....                         | 36 |
| 2.3.3                                 | Hybrid .....                                      | 38 |
| 2.4                                   | Research Gaps .....                               | 41 |
| 2.5                                   | Summary of the Chapter .....                      | 42 |
| Chapter 3: Research Methodology ..... |   | 43 |
| 3.1                                   | Research Methodology .....                        | 43 |
| 3.2                                   | Objectives .....                                  | 45 |
| 3.3                                   | Graphical Representation of the Work .....        | 45 |

|            |   |     |
|------------|---|-----|
| 3.4        | Summary of the Chapter .....  | 48  |
| Chapter 4: | Trust Identification.....   | 49  |
| 4.1        | Introduction .....  | 49  |
| 4.2        | Implementation Design .....   | 49  |
| 4.3        | Summary of the Chapter .....  | 67  |
| Chapter 5: | Labelling and Classification.....                                   | 68  |
| 5.1        | Introduction to the Architecture .....                              | 68  |
| 5.2        | Fuzzy Logic.....  | 77  |
| 5.2.1      | Utilization of Fuzzy Logic in the Proposed Work Frame.....          | 77  |
| 5.2.2      | Root Mean Square Error (RMSE).....                                  | 78  |
| 5.2.3      | Standard Error (SE).....  | 78  |
| 5.3        | Artificial Neural Network (ANN).....                                | 89  |
| 5.3.1      | Working of ANN .....  | 90  |
| 5.3.2      | Implementation Architecture of ANN in MATLAB .....                  | 90  |
| 5.4        | Effect of Training Algorithm Based on Quantitative Parameters ..... | 97  |
| 5.4.1      | Evaluation Using 70:30 Distribution .....                           | 97  |
| 5.4.2      | Evaluation Using 80:20 Distribution .....                           | 100 |
| 5.4.3      | Evaluation Using 90:10 Distribution .....                           | 103 |
| 5.5        | Summary of the Chapter .....  | 106 |
| Chapter 6: | Results and Comparison.....   | 108 |
| 6.1        | Background Information .....  | 108 |
| 6.2        | QoS Effect Based on Different Classifiers.....                      | 108 |
| 6.3        | Comparative Analysis for QoS Measurement.....                       | 115 |
| 6.4        | Summary of the Chapter .....  | 124 |

|                            |     |
|----------------------------|-----|
| Chapter 7: Conclusion..... | 125 |
| 7.1 Conclusion.....        | 125 |
| 7.2 Future Scope.....      | 127 |
| REFERENCES .....           | 129 |
| LIST OF PUBLICATIONS ..... | 142 |



# LIST OF FIGURES

---

|  |    |
|--|----|
| Figure 1.1 VANET Communications [6].....                         | 3  |
| Figure 1.2 VANET System Architecture [5].....                    | 6  |
| Figure 1.3 Architecture for C2C-CC [15].....                     | 8  |
| Figure 1.4 Salient Features of VANET Communication [5, 6].....   | 9  |
| Figure 1.5 VANET Challenges [16].....                            | 11 |
| Figure 1.6 Routing Protocol Classification [7].....              | 12 |
| Figure 1.7 Architecture of RP's [7].....                         | 14 |
| Figure 1.8 Applications Based on VANETs [31-34].....             | 19 |
| Figure 2.1 Routing Protocols [97].....                           | 34 |
| Figure 2.2 The Discovery Process.....                            | 35 |
| Figure 3.1 Graphical Abstract of the Proposed Work.....          | 47 |
| Figure 4.1 Node Deployment.....                                  | 52 |
| Figure 4.2 The Vehicle Distribution.....                         | 53 |
| Figure 4.3 The Communication Network.....                        | 54 |
| Figure 5.1 Data Separation Using Machine Learning.....           | 70 |
| Figure 5.2 Implementation Design for Separation of the Data..... | 71 |
| Figure 5.3 Architectural Input to the K-Means.....               | 75 |
| Figure 5.4 Labelling using MSE.....                              | 76 |
| Figure 5.5 Fuzzy Inference Engine Models.....                    | 79 |
| Figure 5.6 Fuzzy Logic Creation.....                             | 79 |
| Figure 5.7 Input Membership Function.....                        | 80 |
| Figure 5.8 Output Membership Function.....                       | 80 |

|   |     |
|---|-----|
| Figure 5.9 The Rule Viewer.....   | 83  |
| Figure 5.10 Moderate MSE and Moderate SE Resulting into Moderate Closeness..... | 84  |
| Figure 5.11 High MSE and Moderate SE Resulting into Low Closeness .....         | 85  |
| Figure 5.12 Low MSE and Moderate SE Resulting into Moderate Closeness.....      | 85  |
| Figure 5.13 High MSE and Moderate SE Will Result into Low Closeness .....       | 86  |
| Figure 5.14 The Surface View of the Inference Engine .....                      | 87  |
| Figure 5.15 Biological Neuron .....   | 90  |
| Figure 5.16 Structure of ANN .....  | 90  |
| Figure 5.17 Propagation Architecture of Neural Network.....                     | 92  |
| Figure 5.18 Neural Regression Architecture .....                                | 93  |
| Figure 5.19 Regression Change Behaviour .....                                   | 94  |
| Figure 5.20 MSE Validation Scenario 1 .....                                     | 95  |
| Figure 5.21 MSE for Scenario 2 .....  | 96  |
| Figure 5.22 TPR Analysis using 70:30 Data Division.....                         | 99  |
| Figure 5.23 FPR Analysis using 70:30 Data Division.....                         | 99  |
| Figure 5.24 TPR Analysis using 80:20 Data Division.....                         | 102 |
| Figure 5.25 FPR Analysis using 80:20 Data Division.....                         | 102 |
| Figure 5.26 TPR Analysis using 90:10 Data Division.....                         | 105 |
| Figure 5.27 FPR Analysis using 90:10 Data Division.....                         | 106 |
| Figure 6.1 Evaluation of Throughput Based on Different Classifiers .....        | 109 |
| Figure 6.2 Evaluation of PDR Based on Different Classifiers .....               | 111 |
| Figure 6.3 Evaluation of TDR Based on Different Classifiers .....               | 113 |
| Figure 6.4 Evaluation of Jitter Based on Different Classifiers .....            | 115 |
| Figure 6.5 Comparative Analysis of Throughput .....                             | 117 |

|   |     |
|---|-----|
| Figure 6.6 Improvement Analysis of Throughput ..... | 117 |
| Figure 6.7 Comparative Analysis of PDR .....        | 119 |
| Figure 6.8 Improvement Analysis of PDR .....        | 119 |
| Figure 6.9 Comparative Analysis of TDR .....        | 121 |
| Figure 6.10 Improvement Analysis of TDR .....       | 121 |
| Figure 6.11 Comparative Analysis of Jitter .....    | 123 |
| Figure 6.12 Improvement Analysis of Jitter .....    | 123 |

## LIST OF TABLES

---

|  |     |
|--|-----|
| Table 1.1 Comparison between VANET and MANET [4] .....                             | 4   |
| Table 2.1 Comparative Analysis of Approaches for Addressing Attacks in VANETs .... | 39  |
| Table 4.1 RSU Formula Validation .....   | 50  |
| Table 4.2 Simulation Architecture.....   | 54  |
| Table 4.3 Sample Simulation Result.....  | 61  |
| Table 4.4 Response Time.....   | 65  |
| Table 5.1 Affected Discoveries According to K-Means.....                           | 73  |
| Table 5.2 Ordinals of Neural Network .....   | 88  |
| Table 5.3 Performance Analysis using 70:30 Data Division.....                      | 97  |
| Table 5.4 Confusion Matrix for 70:30 Data Division.....                            | 100 |
| Table 5.5 Performance Analysis using 80:20 Data Division.....                      | 100 |
| Table 5.6 Confusion Matrix for 80:20 Data Division.....                            | 103 |
| Table 5.7 Performance Analysis using 90:10 Data Division.....                      | 103 |
| Table 5.8 Confusion Matrix for 90:10 Data Division.....                            | 106 |
| Table 6.1 Evaluation of Throughput Based on Different Classifiers.....             | 108 |
| Table 6.2 Evaluation of PDR Based on Different Classifiers.....                    | 110 |
| Table 6.3 Evaluation of TDR Based on Different Classifiers .....                   | 112 |
| Table 6.4 Evaluation of Jitter Based on Different Classifiers .....                | 113 |
| Table 6.5 Comparative Analysis of Throughput.....                                  | 116 |
| Table 6.6 Comparative Analysis of PDR.....   | 118 |
| Table 6.7 Comparative Analysis of TDR .....  | 120 |
| Table 6.8 Comparative Analysis of Jitter .....                                     | 122 |

## ABBREVIATIONS

---

| <b>Abbreviation</b> | <b>Full Name</b>                          |
|---------------------|---|
| WSN                 | Wireless Sensor Network                   |
| SN                  | Sensor Nodes                              |
| MANET               | Mobile Ad-Hoc Network                     |
| VANET               | Vehicular Ad-Hoc Network                  |
| TMC                 | Traffic Management Centres                |
| OBU                 | On Board Unit                             |
| AU                  | Application Unit                          |
| HS                  | Hot Spot                                  |
| RSU                 | Road Side Unit                            |
| V2V                 | Vehicle to Vehicle                        |
| V2B                 | Vehicle to Broadband cloud                |
| V2I                 | Vehicle to infrastructure                 |
| LBRP                | Load Balancing Routing Protocol           |
| ZOR                 | Zone of Relevance                         |
| ZOF                 | Zone of Forwarding                        |
| BFA                 | Bellman Ford Algorithm                    |
| OLSR                | Optimized Link State Routing              |
| DSDV                | Destination Sequenced Distance Vector     |
| MPR                 | Multi-Point Relays                        |
| DAS                 | driver-assistance technology              |
| ACC                 | adaptive cruise control                   |
| CACC                | cooperative adaptive cruise control       |
| MARS                | machine learning-assisted route selection |
| CSI                 | channel state information                 |
| IVN                 | in-vehicle networks                       |
| D2D                 | Device to Device                          |
| Dos                 | Denial of Service                         |

|       |  |
|-------|--|
| HMM   | hidden Markov model                      |
| VAN   | Vehicular adhoc Networking               |
| ML    | Machine Learning                         |
| DRL   | Deep Reinforcement Learning              |
| RF    | Radio Frequency                          |
| VVLC  | Vehicular Visible Light Communications   |
| RSU   | Remote Sensing Unit                      |
| MAC   | Medium Access Control                    |
| CR    | Cognitive Radio                          |
| ANN   | Artificial Neural Network                |
| V2P   | Vehicle-to-Passenger                     |
| VAN   | vehicular ad hoc networks                |
| GCC   | Global Chassis Control                   |
| VDIM  | Vehicle Dynamics Integrated Management   |
| ADS   | Automotive Doppler Sensing               |
| 5G    | Fifth Generation                         |
| DSRC  | Dedicated Short Range Communication      |
| ML/DL | Machine/Deep Learning                    |
| EV    | Electric Vehicle                         |
| V2X   | vehicle-to-everything                    |
| PR    | Precision Recall                         |
| ROC   | Receiver Operating Characteristic        |
| LSTM  | Long Short-Term Memory                   |
| CNN   | Convolutional Neural Network             |
| DNN   | Deep Neural Network                      |
| SVM   | Support Vector Machine                   |
| QOE   | Quality of Experience                    |
| IoV   | Internet of Vehicle                      |
| IoT   | Internet of Things                       |
| AV    | Autonomous Vehicle                       |
| MLA   | Machine Learning Algorithms              |
| QoS   | Quality of Service                       |
| AODV  | AD HOC ON DEMAND DISTANCE VECTOR ROUTING |

|      |                                       |
|------|---------------------------------------|
| RREQ | Route Request                         |
| RREP | Route Reply                           |
| RRER | Route Error Message                   |
| DSR  | Dynamic Source Routing                |
| TORA | Temporarily ordered routing algorithm |
| OLSR | Optimized Link State Routing Protocol |
| MPR  | Multipoint Relay                      |
| WRP  | Wireless Routing Protocols            |
| MRL  | Message Return Table                  |
| RT   | Routing Table                         |
| DT   | Distance Table                        |
| PDR  | Packet Delivery Ratio                 |
| SE   | Standard Error                        |
| MSE  | Mean Squared Error                    |
| FL   | Fuzzy Logic                           |
| FLS  | Fuzzy Logic System                    |
| FST  | Fuzzy Set Theory                      |
| RMSE | Root Mean Square Error                |
| NB   | Naïve Bayes                           |

## ABSTRACT

---

An intelligent transportation system has a potential to save human lives by safely disseminating the information amongst the moving vehicles. The research “**An Improved Security Algorithm for V2V Communication using Machine Learning**” aims at proposing an algorithm which has the capability to securely send messages in a vehicle to vehicle communication model. An extensive literature review has been presented which elaborates the advancements by the researchers for differentiating by trusted & untrusted vehicles and their evaluation. Various communication based models are analyzed & compared using machine learning. Various QoS like throughput, PDR, Jitter and TDR are used to assess the quality of the proposed algorithms. The research gaps have been studied well and objectives have been drawn according to the analytical survey done during the phase of review of extensive literature proposed and available on the given area.

The major objectives of the research are to device an improved algorithm which efficiently differentiates between the nodes which are trusted and the ones which are untrusted nodes. Further, packet efficient prevention architecture was developed to enhance the security aspect of the communication over V2V, having Machine Learning aspects incorporated to it. In order to implement this, a cluster head based Ad-hoc on demand distance vector algorithm has been generated which uses dragon fly for the optimization on the basis of previous behavior. The concept has been introduced based on the reactive routing in VANET and its implementation has been done in MATLAB. The deployment node, communication architecture, and simulation properties has been described. Ten thousand rounds of simulations have been observed and used for the research. The proposed architecture is such that it addresses more than one attack. K-means clustering has been used to create clusters of the hit data. Fuzzy inference system has been used to identify the three types of attacks amongst the simulated hit data.



The architecture of the proposed work is dependent upon the establishment of the trust via machine learning among the vehicular nodes in order to attain maximum QoS values. Jitter, PDR, TDR and Throughput have been observed in the simulated data and the results are discussed and compared with the existing work of other researchers and significant improvement has been observed. The proposed algorithm has used training and classification architecture, the evaluation of this section has been made based on the quantitative parameters viz. true positive rate and false positive rate. The proposed algorithm architecture has been also compared with other state of art classification algorithms such as Naïve Bayes, Random Forest. Further, the evaluation has been done considering the 70:30, 80:20, and 90:10 distribution ratio.

For 70:30 distribution ratio, TPR and FPR results are compared with the existing techniques such as Naïve Bayes and Random Forest. The average value for TPR and FPR using the proposed technique is 0.96 and 0.064 respectively while existing technique shows TPR of about 0.93 for both Naïve Bayes and 0.95 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.084 and 0.08 respectively. Thus, proposed technique shows prominent FPR and TPR in comparison to Naïve Bayes and Random Forest.

For 80:20 distribution ratios, the average value of TPR and FPR using the proposed technique is about 0.96 and 0.064 respectively while existing technique shows TPR of about 0.93 for both Naïve Bayes and 0.95 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.09 and 0.07 respectively. Thus, proposed technique shows prominent FPR and TPR for 80:20 ratios in comparison to Naïve Bayes and Random Forest.

For 90:10 distribution ratios, the average value of TPR and FPR using the proposed technique is about 0.97 and 0.053 respectively while existing technique shows TPR of about 0.950 for both Naïve Bayes and 0.958 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.079 and 0.071

respectively. Thus, proposed technique shows prominent FPR and TPR using 90:10 ratio in comparison to Naïve Bayes and Random Forest.

For QoS measurement, throughput analysis using the proposed approach and the existing techniques has been presented. It is clearly visible that the average throughput value by Yuvraj et al. and Kim et al. is 18969.4 kbps and 18545.4 Kbps respectively. The proposed technique shows average throughput of about 19219.39 kbps. In comparison to the Yuvraj et al. and Kim et al., the improvement of the proposed approach is 1.32 % and 2.42 % respectively

Further, PDR analysis using the proposed approach and the existing techniques has been presented. It is seen that PDR using the proposed approach is 0.94 while PDR with Naïve Bayes Gaussian is 0.91. Consequently, PDR using the Random Forest is 0.93. Thus, PDR using the proposed technique is much more efficient than the previously existing mechanisms. The improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 4.44% and 6.6%.

TDR observed using proposed work is 94.65% while the TDR exhibited by Yuvraj et al. and Kim et al. is 89.016% and 90.86%, respectively. The TDR improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 6.5% and 4.3%.

The Jitter value for 100000 nodes by Yuvraj et al. and Kim et al. is 31.51 and 29.79 respectively. The improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 16.92% and 11.24% respectively. The improvement analysis shows the robustness of the proposed technique.

## Chapter 1: Introduction

---

VANETs are robust and multidisciplinary networks used for communication and powered with the its potential as a result of technological advancements in the transportation sector (VANET). This chapter is dedicated to introducing the basic concepts, techniques, and routing protocols that form the basis of communication in VANET.

### 1.1 Vehicular Ad-Hoc Network (VANET)

In VANET, nodes are used to monitor the environmental conditions such as temperature, sensitivity, sound, humidity, and direction. The concentration of nutrients in the air is also determined along with the pollutants and many other levels [1]. The sensor devices can be equipped with actuators, allowing them to "act" in reaction to specific circumstances [2]. VANET is used in a variety of projects, such as Chrysler etc. Communication is possible spontaneously and wirelessly through various technology vehicles. With the use of intelligent transportation technologies, travellers can travel in greater safety and comfort. Many academics and industry researchers are interested in VANET communications. As long as, at least one of the sending and the receiving units is a vehicle and may be a routing node, it is described as communication between the vehicles and possibly with the road side units [3]. Mobile ad hoc networks (MANETs), particularly vehicular ad hoc networks (VANET), become increasingly common. MANET is a part of VANET which is a collection of SNs used to connect the user without any central connection through the network and it generally has enough information and all the capabilities to connect with the network. In the case of VANET, it is difficult to make a reliable and feasible connection with the variation of nodes. Additionally, different RP's have been used to make the network error-free and load free [4]. This inter-vehicle communication allows the data to be passed back and forth, improving the efficiency of the traffic, detecting road conditions, reducing the number of crashes, detecting emergency situations, and overall efficiency of the network. With the

use of multi-hops, VANET also transmits data to distant devices [5]. This work presented comparison and contrast between the VANET and MANET environments, as well as the protocols used in these two sectors.

## 1.2 Vehicular Networks (VANETs)

The notion of VANET has gained popularity, allowing for new applications to be used in the field of safety. VANET has various applications used in distinct fields and is known for moving vehicles and used for other connecting equipment. Communication is possible with one another and it shares crucial information through a wireless medium. A small network is developed at the same time, with vehicles and other devices acting as network nodes. The nodes have the ability and are equipped with enough battery power to send any information they have to the other nodes. Similarly, all nodes associated with and indulged in the network have been used to receive the data sent by other nodes after delivering their own set of data. Nodes work to generate useful information from this data after gathering it all, which they then broadcast to other devices. [6-7].

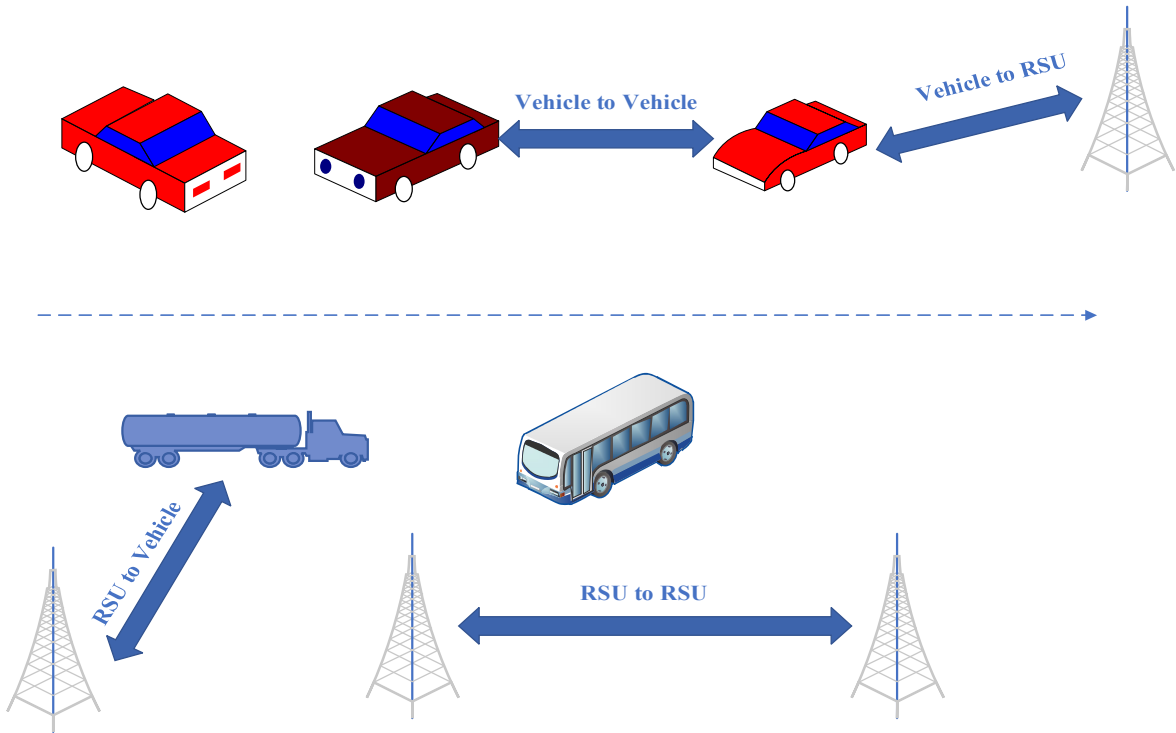


Figure 1.1 VANET Communications [6]

It is an open-source network in which nodes have enough ability to join or connect and disconnect from the network at any time. On-board sensors have been used for connection purposes in which VANET's benefits have been accessed by connecting and merging into the network.

### 1.2.1 Concept

Although VANET is a MANET application, it has its own special properties that can be summarised-

- High Mobility: VANET nodes are typically having high speed are used to send the data with efficient capability. It is more difficult to predict the position in the network where the position of the vehicle node is changing rapidly [8].
- Network Topology: The position of the node changes frequently due to high node mobility and erratic vehicle speeds.

- Size of the Network (unbounded): The set-up of a VANET has been designed in a single city, a group of cities, or even a whole country. As a result, the VANET network's scalability is not constrained.
- Exchanging the information on a frequent basis: VANET encourages nodes to collect data from other vehicles and roadside equipment. As a result, node-to-node communication becomes more frequent. [9].

### 1.2.2 Comparison of VANET and MANET

The important difference between the VANET and MANET is the behaviour and sharing properties. The other one is the cost of initial production when compared to MANET; the VANET production cost is higher. The VANET network topology is frequent, rapid, and mobile due to the high speed of cars, but the MANET network topology is sluggish and slow. In comparison to Mobile Ad-hoc networks, VANET has a higher bandwidth. In MANET, nodes move at random, whereas in VANET, nodes move in a predictable pattern.

Table 1.1 Comparison between VANET and MANET [4]

| Sr. No. | Components              | VANET                            | MANET                              |
|---------|-------------------------|----------------------------------|------------------------------------|
| 1       | Production Cost         | High                             | Inexpensive                        |
| 2       | Network Topology Charge | Frequent and very fast           | Slow                               |
| 3       | Mobility                | High                             | Low                                |
| 4       | Density in Node         | Frequent Variable and Dense      | Sparse                             |
| 5       | Bandwidth               | Thousand Kps                     | Hundred Kps                        |
| 6       | Range                   | Up to 600 mtr                    | Up to 100 mtr                      |
| 7       | Node Lifetime           | It is Depended Vehicle Life time | It is Depended on the Power Source |
| 8       | Reliability             | High                             | Medium                             |

|   |                      |         |        |
|---|----------------------|---------|--------|
| 9 | Nodes Moving Pattern | Regular | Random |
|---|----------------------|---------|--------|

### 1.3 Components of VANET

In this section, the components of VANET have been discussed. The vehicle-to-vehicle communication and information transmission using the node locations and routing tables has been detailed. The interaction using the communication architecture and VANET components has been detailed.

#### 1.3.1 Important Component

The vehicles have been categorized into three domains viz Mobile, Generic, and infrastructure, in which the first one uses the vehicle as a mobile device. The second one uses the internet and private network for connection and the last one is the central connection for infrastructure and roadside interconnections.

The vehicle area and the cell phone space are the two components of the versatile area. A wide range of convenient gadgets, like individual route gadgets and cell phones, are widely used for the data transfer in the specified domain [10-12].

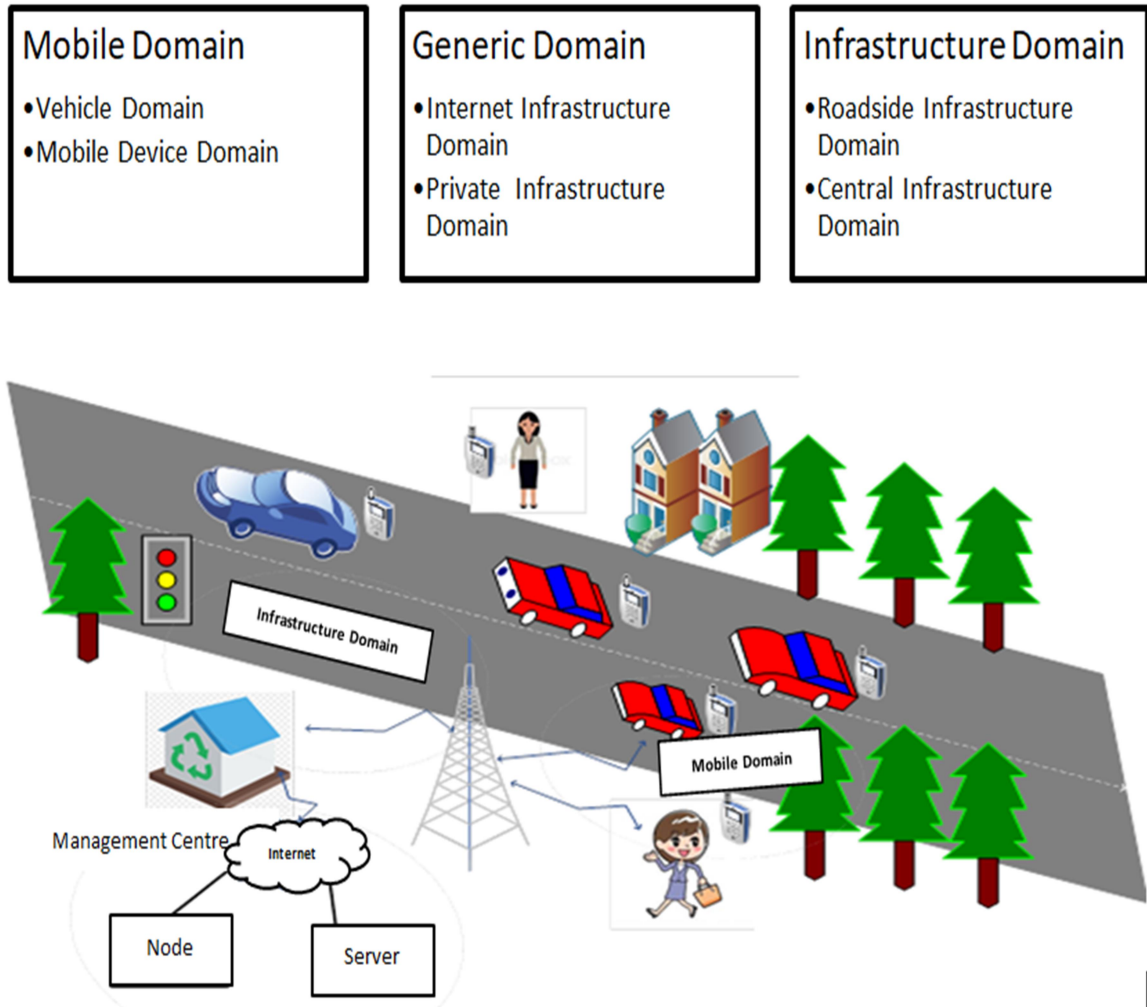


Figure 1.2 VANET System Architecture [5]

The side of the road foundation area and the local framework space are two areas inside the infrastructure domain. In other words, traffic signals have been used and other side-of-the-road unit substances are important for the side-of-the-road foundation area. The central domain is specified as an infrastructure domain that contains a framework for the board communities, for example, traffic management centres (TMCs), and are also used for vehicle management centres [13]. Each of these components is having their own utility and specifications and collectively contributes to the formation of VANET architecture



- Mobile domain: The vehicle domain and mobile device domain are the two main components of the Mobile Domain. The vehicle domain is used for moving vehicles that includes any car, truck, etc. and the second domain is for mobile devices which include laptops, cell phones, and so on. In today's era, with the extensive span of mobile devices, this category stands as the most prominent and most potential for research purposes.
- Generic domain: It belongs to the infrastructure that makes the network private and contains several nodes and interconnecting devices that worked in a VANET for indiscipline computing resources. This can be categorised into two sub categories each having their own specifications. The first one is Internet Infrastructure Domain, which spans the entire internet. Whereas the second one is Private Infrastructure Domain which deals with the privatised network systems.
- Infrastructure domain: There are two components to it as well. The first one is roadside infrastructure domain and the second is central infrastructure domain. This includes roadside elements and other infrastructure required to establish the transfer.

The mobile domain communicates and exchanges data with the other domain to analyse and manage the information to perform certain actions. The infrastructure domain then talks with the generic domain and shares information with it in the second stage. This data flow between stationary and mobile resources allows users to make more efficient and productive use of the route.

The on-board unit (OBU) and application units belong to the in-vehicle domain as shown in Figure 1.3. The connections are used to connect the devices using linkages that can be wired or non-wired. The system is made up of different vehicles used to link through a static node and a dynamic node. A wide range of convenient gadgets, like individual route gadgets and cell phones, are widely used for the cell phone in the specified domain using the static node. The linking of the RSU using the cellular network and OBU has been done to communicate the information and linkages between the nodes [14].

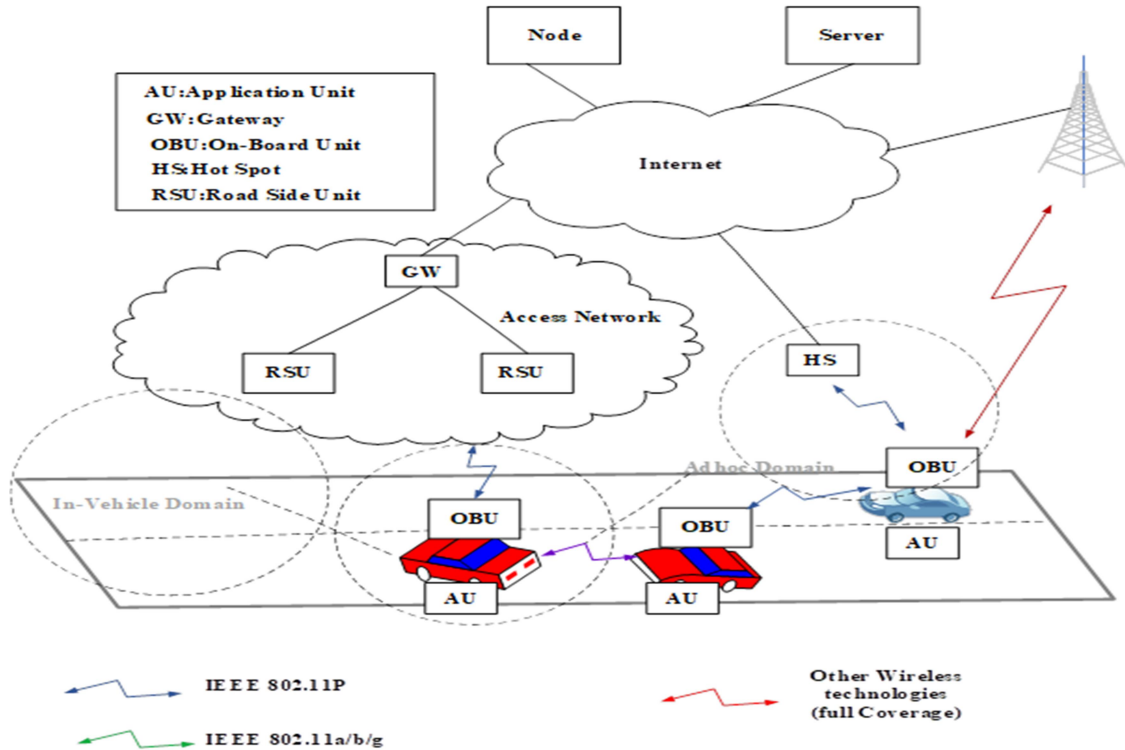


Figure 1.3 Architecture for C2C-CC [15]

### 1.3.2 Types of Communication

The communication architecture of the VANET has been categorized into four different types [6] which are depicted as follows:

#### 1.3.2.1 In Vehicle Communication

This type of communication is used to measure the performance of the system such as a driver's driving a car and there is a drowsiness that is intense. The identification of such signs indicates that magnitude is very critical and intense for both driver and the safety of the vehicle.

#### 1.3.2.2 Vehicle to Vehicle (V2V)

The transmission process between the vehicles has been carried out in order to help and assist the drivers in case of emergency and provide essential information. Nevertheless, it

does not require a fixed infrastructure for data sharing and is useful for applications such as data dissemination, safety, and security.

### 1.3.2.3 Vehicle to Broadband Cloud (V2B)

It is used for communication purposes using the vehicle and there is a need for broadband devices to share the information and different networks such as 3G or 4G that has been used for determining the statistics of the traffic, this improves driver assistance and vehicle tracking.

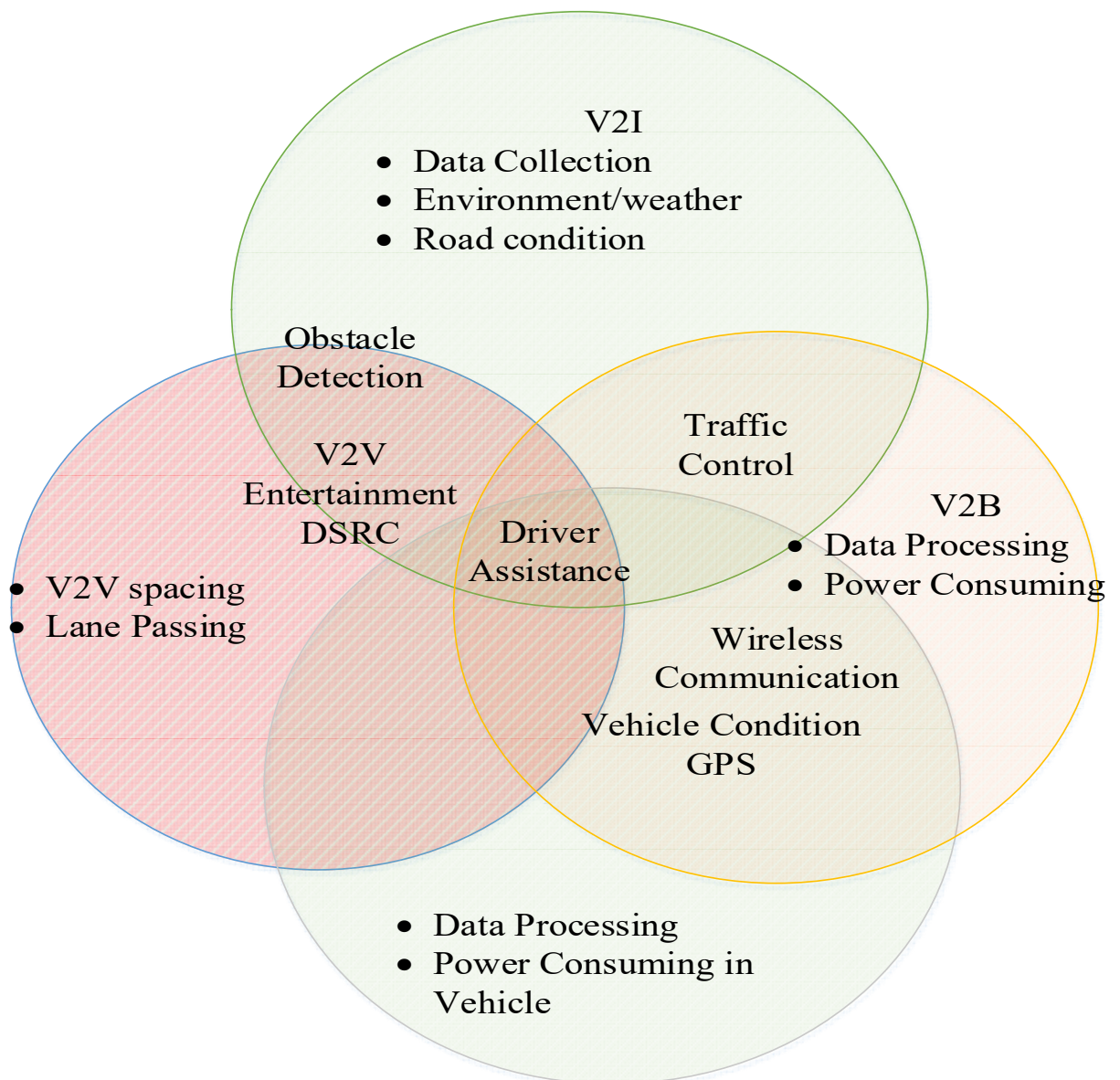


Figure 1.4 Salient Features of VANET Communication [5, 6]

#### 1.3.2.4 Vehicle to Infrastructure (V2I)

This communication takes place along the roadside in order to equip with the information for communication from one place to another. It is widely used for delivering protective information and others such as real-time information.

All of the communication kinds outlined above have taken place in VANETs architecture. Information sharing begins as vehicles move to transmit the information as long as the vehicle remains in the VANET; it functions and benefits from it

#### 1.4 Challenges in VANETs

Traditional VANETs suffer different issues and challenges related to the management and technological devices along with flexibility issues and scalability. However, poor connection and lack of intelligence devices are added challenges suffered during the communication process. There are a number of technological issues that must be addressed prior to the deployment of VANET [16]. Some challenges are given below:

1. Network Management: The system is based on the network topology and it is the connection with channel state that varies rapidly due to high mobility. As a result, tree structures can't be employed because of the change in topological infrastructure.
2. Congestion and Collision Control: A problem is also posed by the network's infinite size. Rural locations have low traffic loads, and even urban areas have minimal traffic loads at night. As a result, network partitions are common during rush hours, when traffic loads are extremely high it causes the network to become congested and forces collisions to occur.
3. Environmental Impact: Electromagnetic waves are used to perform communication in VANETs. The environment has an impact on these waves. As a result, the environmental impact of deploying the VANET is added.
4. MAC Design: A MAC address, short for Media Access Control address, is a unique identifier assigned to a network interface card by the manufacturer. It is a hardware address that is used to uniquely identify devices on a network at the data

link layer of the OSI model. It is unique and has distinct phases to connect with the other devices.

5. Security: As a result of VANET's provision of life-saving road safety apps, the security of these messages must be ensured.

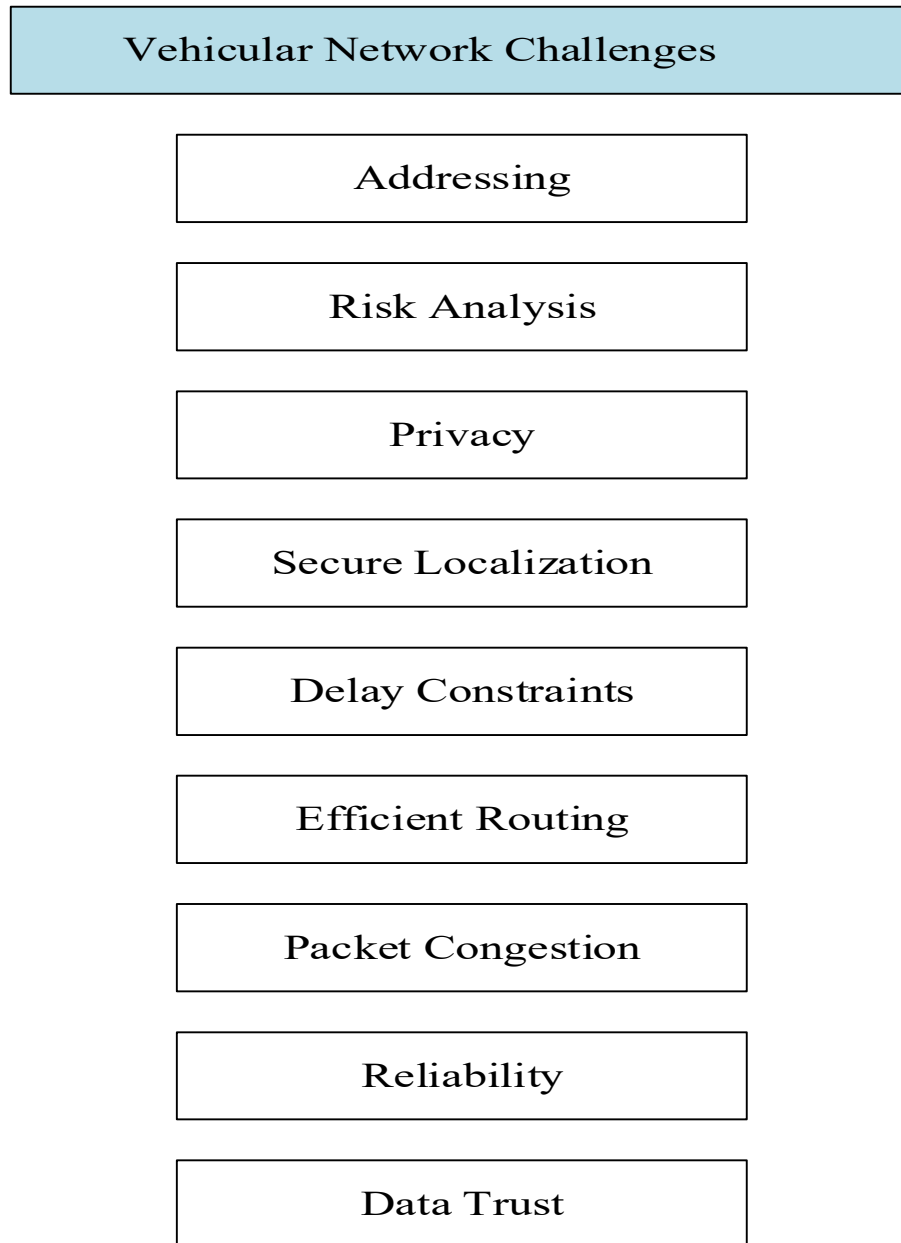


Figure 1.5 VANET Challenges [16]

## 1.5 Routing (routing protocols and their challenges)

Due to the high mobility of VANET, choosing the right routing protocol is crucial. The packets in the network have been transferred from one device to another using the number of cars which are increasing and decreasing, posing new issues for routing protocols. Also, because of the complex nature of VANET, the practitioners used different protocols as described in the below sections [7]. In addition, Figure 1.6 depicts the routing protocol classification.

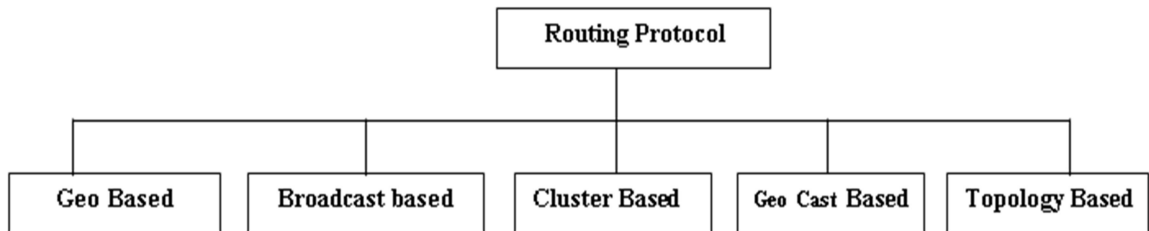


Figure 1.6 Routing Protocol Classification [7]

### 1.5.1 Protocols Based on Geographical Information

The geographical locations have been used to connect the end-to-end communication using the network address in these protocols. The Load Balancing Routing Protocol (LBRP) calculates and configures routes based on information from node locations. As a result, there is no need to create routing tables. Beacons, position, and service of transferring the information are the main protocols of geographical positioning. Furthermore, as the car travels through a region such as a tunnel, satellite signals become poor. However, when it comes to the highway environment, it performs well. It also has the benefit of being efficient and works well in a complex system.

### 1.5.2 Broadcast Based Routing Protocols

This protocol sends data packets to the neighbour nodes using the broadcast channel across the entire VANET. When the service of a particular node is out of range or inefficient, the said protocol is applied. Typically, different condition has been employed to warn the users against safety protocols such as conditional alarms, warning systems, and so on. There are different examples of broadcasting routing protocols for instance

Decentralized Congestion Awareness (DECA) based on position, density, etc. The reliability of the aforementioned procedures is a plus. However, these protocols use more bandwidth, and a large number of packets that are redundant reach the destination which badly impacts the performance of the network.

### 1.5.3 Cluster Based Routing Protocols

Vehicles having similar properties are grouped in terms of energy, speed and so on under this protocol. Furthermore, communication via local node is possible using the vehicle nodes that are communicated using the cluster node through a direct path. Furthermore, communication is possible when the interaction has been done with the usage of Cluster Head (CH), it will need the assistance of its (CH) to get there. It is a fantastic choice for network builders because of the scalability factors. However, one of its disadvantages is traffic congestion. This protocol is best shown by an open interaction network in which communication is possible through the clustering process.

### 1.5.4 Geo-cast Routing Protocols

The Zone of Relevance (ZOR) and Zone of Forwarding (ZOF) are the two key parts of the Geo-cast protocol. The former part is used to reserve the area for the regional nodes. The major objective of this protocol is to allow vehicles in ZOR to communicate with one another. Furthermore, the communication between the vehicles has taken place when the interaction is especially within the ZOR rather than ZOF on a regular basis. The communication between the nodes has been carried out considering the ZOR, and this point falls into the category of downsides.

### 1.5.5 Routing Protocols Based on the Topology

This section throws light on the RP's based on the topology. The three types of protocols are there such namely reactive, proactive, and hybrid. Figure 1.7 also shows the parts of RP's that interacted with the nodes in an interactive environment.

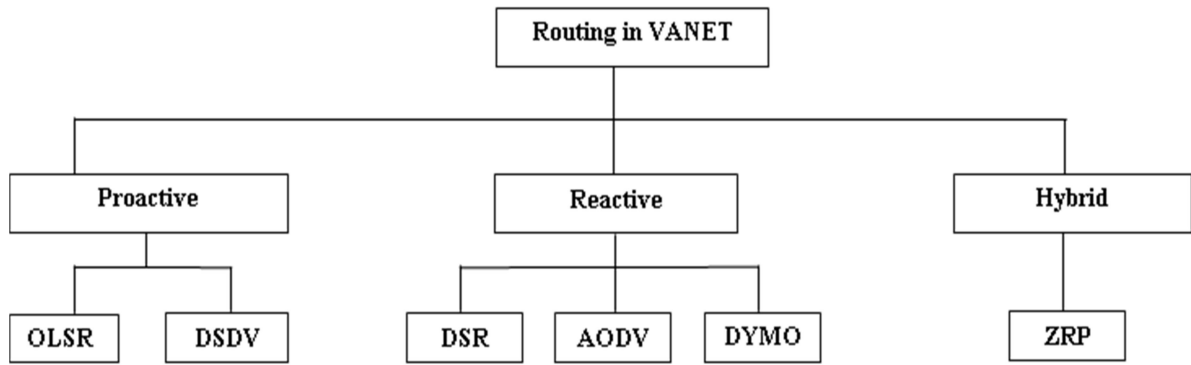


Figure 1.7 Architecture of RP's [7]

### 1.5.6 Proactive Protocols

The protocols where the route is decided before the actual data transfer come under this category. Because the protocols compute the route on a regular basis, the routing table changes or updates often. The Bellman-Ford Algorithm (BFA) is a well-known protocol that is used to keep the data and information safe and thus, transfer the information further by establishing robust routes. Optimized Link State Routing (OLSR) and Destination Sequenced Distance Vector (DSDV) are two examples of these protocols.

The OLSR protocol is essentially a more polished and improved form of the link state protocol. This protocol works in such a manner that changes in the topological network broadcast the messages to all the connected nodes, thus increasing the cost of the network. The Destination Sequence Distance Vector (DSDV) RP works on the basis of an algorithm of Bellman-Ford. The information is in the form of a sequence number that has been maintained to avoid the looping problem in the routing architecture.

#### 1.5.6.1 Destination Sequenced Distance Vector (DSDV) Routing

The goal of the DSDV protocol is to construct routes that are loop-free and available while using end-to-end communication and the technique based on distance vector is also used to determine the shortest path. The protocol sends two sorts of packets namely, the incremental and the other is full dump. The routing information is bestowed in full dump packets and the updates are present in the incremental packets. Full dump packets consume more bandwidth, while incremental packets are sent more frequently, increasing



network overhead. Due to bandwidth usage and update methods, the DSDV protocol is not appropriate for large networks [17].

#### 1.5.6.2 Optimized Link State Routing Protocol (OLSRP)

OLSR is a robust protocol used for point-to-point communication using the traditional RP that falls under the category of proactive. For route establishment or route maintenance, it employs a technology known as multipoint relaying, which optimizes the message and flooding process. Multi-Point Relays (MPR) [18] is an algorithm that reduces the number of active relays required to cover the neighbours. The protocol was created to ensure the accuracy and stability of data routing in a network. Neighbours such as optimized state and multipoint relays are well-established protocols that maintain the routing information and two-hop links. These are two fundamental principles in the Optimized Link State Routing protocol (OLSR). Updates are only received once per node, and unselected packets are unable to retransmit updates.

#### 1.5.7 Protocols Based on the Reactive Nature

Such protocols have no access to all of the nodes' information [19]. It simply saves information about the nodes that pass via the route. AODV, DSR, and DYMO are examples of reactive protocols.

In VANET, if the information about the route is present in a table, it will be forwarded to the destination. If this is not the case, the overhead of broadcasting is decreased, thanks to the protocols mentioned. Furthermore, as per [20] the route discovery procedure would be done on demand. After AODV, DYMO work in two modes [21], and route finding approach is available on demand anytime.

##### 1.5.7.1 Dynamic Source Routing (DSR)

The working of Dynamic Source Routing is similar to AODV in the sense that it also creates the routes on demand rather than tables to do so. DSR does not use beacons and does not require hello packets to be sent on a regular basis. DSR works by flooding route request packets into the network dynamically, then having the destination node respond

to carry the route-traversed packet in its header. The complete order list of nodes allows packets to be routed, eliminating intermediary location checking. Using this technology, a route is now included to access the information about the packets transferred from one location to another in static mapping [22].

#### 1.5.7.2 Ad-hoc On-Demand Distance Vector (AODV)

AODV creates routes on demand and relies on source routing rather than tables to do so. The AODV protocol is a demand-based reactive protocol. DSDV and DSR algorithms are used in AODV protocols. The discovery process has been initiated using these protocols in which routing tables had been used and requests are processed based on the demand of the user. Further, the process is reached in the discovery process until all the request has been met successfully. Before transmitting the packet to their neighbours, the RREQ packets are unaware of the active route for the requested target. [23].

#### 1.5.8 Protocols Based on Hybridization

It is a combination of two or more protocols working effectively to reduce overhead and energy consumption by sharing topological information on a regular basis [24]. The network's efficiency and scalability have improved, thanks to the hybrid approach. The disadvantage of the hybrid technique, on the other hand, is the high latency when navigating new routes. ZRP is a typical protocol that uses a hybrid method (ZRP).

##### 1.5.8.1 Zone Routing Protocol (ZRP)

This protocol is used to reduce the extra overhead and late delivery of packets associated with the discovery of the route. Furthermore, this is used to split a collection of nodes into zone distinct and overlapping zones, and the nodes are present within the zone radius. The zones are created based on hop distance and selected based on the topological distribution of nodes. The nodes at the zone's edge are known as peripheral nodes. The radius of length determines the size which is accomplished via a protocol that falls inside the zone [25].

## 1.6 Applications of VANETs

Various applications of VANETs are discussed in terms of road safety applications, traffic, and environmental control.

### 1.6.1 Traffic Application for Safe Road

The VANET is primarily used to decrease traffic accidents that result in the death of passengers. These apps provide information and assistance to drivers by exchanging data (intersection position, speed, and distance) between automobiles and RSU in order to avoid crashes with other vehicles and detect dangerous locations via exchanging information. As illustrated in Figure 1.8, some application models are given below [26-28].

- a. Accident due to crossing and vehicle intersection: There is a chance of risk of the collision of the two vehicles which results in intensive damage to the humans.
- b. Support mismatch due to alteration of the lane: It is a common phenomenon on the road to witness vehicles changing lanes. But, not many riders do it carefully considering the person coming from the other side. In case of support mismatch, there is a severe danger to the vehicles that are changing paths from one side to the other.
- c. Overtaking vehicle: There is a risk in a mode such as a bypass in which a vehicle overtakes the other vehicle without giving any prior information to the other driver. For example, there is a risk of vehicle 3 damage as vehicle 1 overtakes vehicle 2 but vehicle 3 without any sign overtakes both, then an accident assure in such a place that results from vehicle crashes.
- d. Emergency vehicle: There are different vehicles that have emergencies such as ambulances, that can pass information to the adjacent vehicles to give way to the emergency vehicles. Such useful messages can be replayed by other vehicles or RSUs.

- e. Taking the wrong direction: There is a probability that a car passes from one to another place using an erroneous direction that leads to serious accidents and crashes.
- f. Conditions (traffic): The vehicles used are driven by the driver in a very fast mode rather than following the speed limits prone to accidents and crashes.
- g. Omitting the laws: RSU is feasible to detect and monitor the violation of signals by collecting the information about the vehicles omitting the signal violation in a hurry.
- h. Risk of collision: There is a risk of collision during the movement of vehicles. So, RSU is feasible and viable to transfer the information to the neighbouring cars. There is a need to follow the traffic rules and need to improve the flow of traffic and control the speeds of the cars. Some models of applications are in [29-30].

#### 1.6.2 Applications for Informative Purposes

“Information is a key resource for successful decision-making” says Bill Gates. The transfer of correct information at right point of time can lead to great achievements. Information sharing stands as one of the most significant applications of VANET. While driving on road if the information regarding the road quality, traffic and the status of congestion is shared securely and timely it can result in saving time, energy and resources.

It can be broadly classified as:

- a. Services supported locally: The information used to convey a viable message and accessed locally by e-commerce [31-32].
- b. Services supported globally: The information via the internet and using different applications such as Skype and other platforms to manage the financial filings and other support globally [33-34].

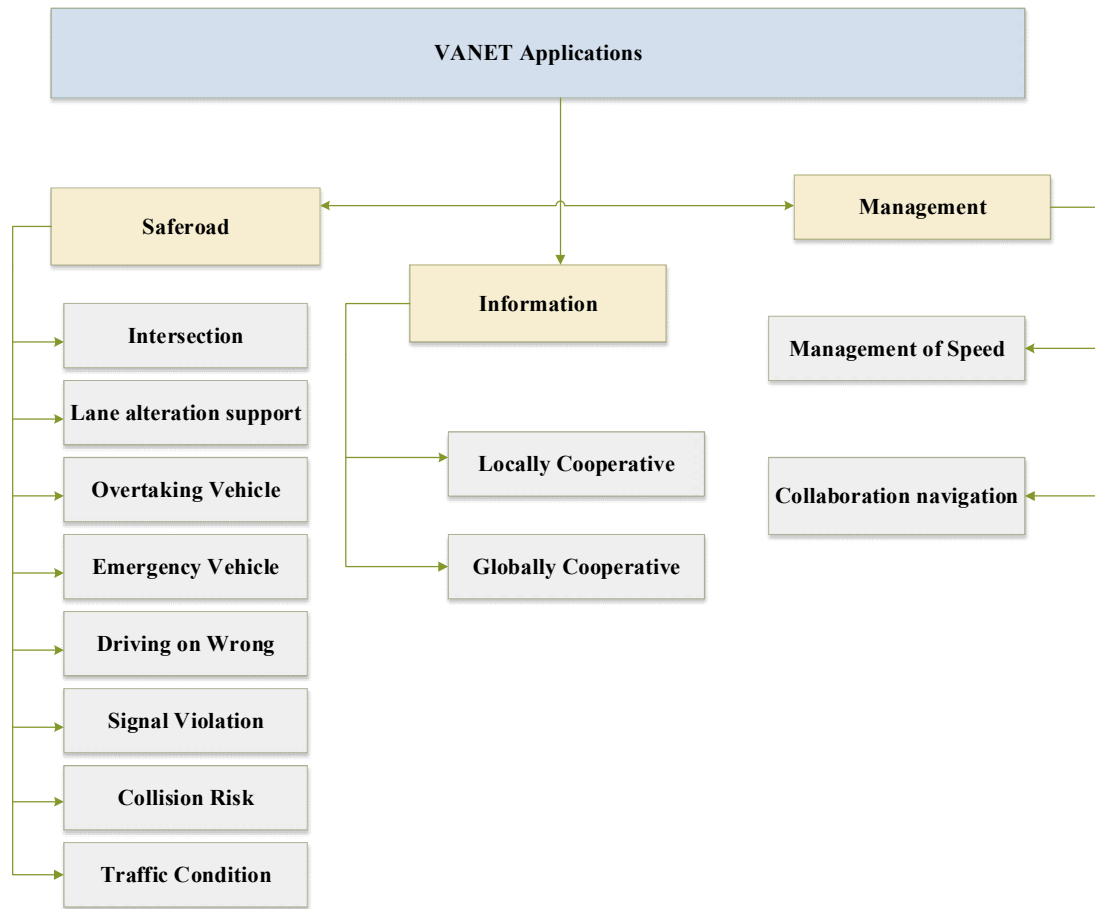


Figure 1.8 Applications Based on VANETs [31-34]

### 1.6.3 Attacks and Threats

It has been observed that a vehicle can cheat the designed security system with its position to escape the liabilities. Thus, a secure position verification of the vehicle holds great significance. Before discussing the behaviour of the type of threats posed by the VANETs, the attackers need to be classified for addressing the threat.

#### 1.6.3.1 Categories of Security Threats in VANETs

The attackers can be broadly classified based on the capacities of the attack [35-37].

- **Insider or Outsider:** Within the network under study, the insider is the legitimate member of the defined network that can communicate with other members of the network. While the outsider is considered an intruder by the network members that limits the attacks that can be mounted by the outsider.

- Active or Passive: The attacker is referred to as an active attacker when it can generate signals or packets while the passive attacker is a type of eavesdropping in the network.
- Rational or Malicious: The attacker is termed a malicious attacker when the attacker is aimed at causing harm to the network members to alter the functionality of the network. While the rational attacker has some motive behind the attack and is usually predictable.
- Local or Extended: Some attackers have limited scope despite the fact that they can control several entities. Such attackers are called local attackers. In contrast to this, the extended attacker can control a number of entities that may be scattered over the network. This extends the area and scope of an attacker and hence is named an extended attacker.

Security is no doubt a key aspect of any communication system and knowhow of the attacker is very important to design security solutions.

## 1.7 Machine Learning

Two of the most common approaches to machine learning are supervised and unsupervised learning. Both options enable the user to supply the computer with a vast amount of data records in order for it to understand and establish relationships. This collected data is commonly referred to as a “feature vector”. There are two categorizations in supervised learning viz. regression and classification. The essence of their production vector is the distinction between the two. If the output variable is in the form of a real value, it is referred to as regression. For example size, weight, height, economic value, etc. Classification, on the other hand, is when the output variable is in the form of a class or category. Labelling the input data into exactly two categories is a binary classification. Marking in more than two classes is a multi-class classification. In the Unsupervised ML technique, the data is not labelled. The machine must find the correct target without any prior knowledge, and therefore detect unknown patterns in the data. For this reason, algorithms must be designed in such a way that they can find the appropriate patterns and structures for themselves in the data. In recent times a tremendous amount of research has been presented by researchers for reward-based

analysis. This type of learning involves reinforcement learning for improving the performance of VANET-based communication architecture [38-39]. A more detailed survey based on machine learning techniques is presented in the next chapter.

## 1.8 Organization of the Thesis

Chapter 1 provides the background of VANETs, important routing protocols, potential attacks, and the impact of machine learning in improving various applications of VANETs.

Chapter 2 summarizes the detailed literature survey conducted while studying the status of existing research in relation to VANETs.

Chapter 3 discusses the problem formulation and the drawn objectives of the current research.

Chapter 4 is dedicated to the identification of the trusted and untrusted nodes in V2V communication while describing the implemented design.

Chapter 5 outlines the packet-efficient architecture aimed to enhance V2V communication. Here, fuzzy logic is utilized for designing the rule set.

Chapter 6 is dedicated to the evaluation of the designed architectures in terms of QoS namely throughput, packet delivery ratio, true detection rate, and jitter.

Chapter 7 finally draws conclusions and discusses the scope of future research.

## 1.9 Summary

In this chapter, VANET is introduced along with its components, and type of communication. Further, the architecture of routing protocols has been detailed and each protocol is explained well. This is followed by the applications of VANET such as safe road applications. The threats and attacks in VANET are also categorized and finally, the organization of the thesis is detailed.

## Chapter 2: Literature Review

---

This chapter is dedicated to the detailed analysis of the existing and past research pertaining to the field of VANETs aimed at enhancing V2V communication. It has been observed that despite numerous applications such as traffic management, data communication, dissemination of warning messages, providing information on real-time road conditions, and diversions to impart driver safety, sometimes there also arise some security concerns. The literature review summarized in this chapter covers the success stories of various researchers available in authenticated publications to lay the foundation of the present research work.

### 2.1 Survey of Existing Research

(Kulkarni et al, 2010) VANETs have distinct applications used to model the problems related to scalability and others. A simulation-based study is critical in achieving the successful deployment of cars in actual circumstances and its associated influence on routing protocols in order to understand the complex problem. The author uses appropriate mobility measures to assess the performance of different mobility models and tries to link their influence on routing protocol performance. In the construction of these sorts of networks, routing protocols are crucial. The authors investigate the applicability of a reinforcement learning-based routing system to fulfill the demanding needs of vehicle networks [38].

(Desjardins, 2011) studies presented an increased capability in sensing, communication, and processing technologies have resulted in the development of driver-assistance technology (DASs). Systems like this are designed to assist drivers by either offering a warning to lessen the likelihood of a collision or performing part of the control chores to relieve a driver of repetitive and tedious jobs. For example, adaptive cruise control (ACC) is designed to relieve a driver of the responsibility of manually altering the speed in order to keep a steady speed or a safe distance from the car. Right now, improving the performance of ACC may be accomplished by vehicle-to-vehicle communication, in



which the current speed and acceleration of a vehicle can be relayed to the following cars through intervehicle communication. As a result, communication between vehicles and adaptive cruise control may be merged into a single system known as cooperative adaptive cruise control (CACC) [39].

(Dogru, 2012), the study used the person's signature behaviour to identify biometric activity. Authentication and authorization are two of the most common uses for it in legal documents. Dynamic or online recognition and static or offline recognition are two methods of verifying a signature. By using Artificial Neural Network, the author analyses signature photos using the offline recognition technique. The author employed machine learning using a supervised learning algorithm for offline signature recognition. The goal of employing an artificial neural network is to identify signatures that match the owners of the signatures automatically. Based on the present study, it was shown that cascade-forward had the maximum accuracy of 100% and the lowest mean square error of 0 when compared to feed-forward back propagation [40].

(Bento, 2013) The legacy algorithm allows the intelligent intersection to handle cars that are used for Vehicle to Vehicle communication in a small number of cases. The created system approaches are designed to reduce the chances of failure, as a result, road traffic's environmental costs. Three intelligent traffic management algorithms are evaluated for use at road junctions, roundabouts, and crossroads. The experimental outcomes show that the proposed system was efficient and reduced the chances of failure and congestion by managing the output flux thus when compared to traditional traffic management techniques, the traffic rate was comparatively very low. When traffic is high, this drop is much more noticeable [41].

(Lai, 2015) Because of the great mobility of cars, the network architecture and communication connections in VANETs are subject to rapid modification. One of the most difficult tasks is determining how to reduce transmission delays while simultaneously increasing transmission stability. The authors used to detect the traffic by selecting the congested routes and vehicles having higher densities that can transmit the packets, and thus focussed on reducing the transmission delay, and is the focus of most

research in routing path establishment. The paper has presented the machine learning-assisted route selection (MARS) system, which estimated the information required by routing protocols to improve their performance [42].

(Perfecto, 2017) The author proposes a unique method for a wireless communication system. The channel state information (CSI) and the queue state information (QSI) were taken into account together while constructing vehicle-to-vehicle (V2V) connections to accomplish this goal. The simulation analysis proved the success of the proposed framework in terms of throughput and latency/reliability trade-offs, as well as how it compares to numerous baseline techniques recently provided in the literature, in order to validate the proposed framework. Results from the proposed research demonstrate that ultra-dense vehicular scenarios may achieve performance benefits in terms of reliability and latency of up to 25%, with an average of 50% more paired cars than some of the baseline scenarios [43].

(Peng, 2018) Vehicle communications, manually driven cars, and autonomous vehicles all have the potential to gather valuable information that may be used to enhance traffic safety and support infotainment services, among other things. From the standpoint of the network layer, the author presents a complete summary of current research on allowing effective vehicular communications from a variety of perspectives. First, the author discusses the basic uses and distinctive properties of vehicular networks, as well as the categories that are used to categorize them. The authors first distinguish between manual and automated systems and then highlight the different techniques used for hand-off and structural networks that can be used in these vehicular networks. The authors then discuss the handoff strategies that can be used in these vehicular networks [44].

(Sim, 2018) Using fast machine learning (FML), the author offers a lightweight, context-aware online learning method with demonstrated performance bounds and assurance of convergence. FML makes advantage of incoming data in order to learn about and adapt to its surroundings. Also, show that a real-world implementation of FML is feasible by offering a standard-compliant protocol that takes use of the current architecture of cellular networks and the capabilities that will be available in 5G in the future. Also

included is the ability to conduct a thorough review using actual traffic patterns collected from Google Maps. Using FML, there is a probability to access the information related to deployment time and it is provided with a learning rate, according to the results of the assessment [45]. Furthermore, by responding quickly to system changes, FML maintains performance within 5% of its ideal level of performance (i.e., blockage, traffic).

(Uchida, 2018) With the advancements, it is believed that communication using the V2V will become more important for new types of distinct applications. However, cars that are generated by the rapid motions of the vehicles or noisy vehicles may create severe problems. When using a technique to adjust the effect of a noisy vehicle in its path, which in turn influences antenna direction. The research was focussed on using the control system that was implemented for direction controls that were developed based on the experimental findings. The results demonstrate the efficacy of the handoff strategies that can be used in these vehicular networks, which is encouraging [46].

(Yuvraj et al. 2018) had presented a cross-layer approach that could handle link asymmetry in order to maintain the quality of service. He has linked the quality of service with congestion avoidance or congestion control-aware routing. The simulation analysis evaluated the effectiveness of the proposed work in terms of reduction in the end-to-end delay with minimal overheads and better throughput. The work successfully defended the maintaining the quality of service to overcome the service deteriorating condition which was congestion in this case. However, the major drawback of this work was that only 40 nodes were considered for the analysis of the congestion control mechanism in a network size of  $1500m^2$  [47].

(Atallah, 2018) Because of the rapid development of the IoT, traditional systems had been evolved to use vehicles using the internet facility. Because of the fast growth of processing, the IoT offers enormous economic interest as well as significant academic value, drawing various advantageous alike that ensure the safety of drivers and the need for continuous connection to the vehicles connected via the internet. The proposed technique solves complex problems in an environmentally friendly manner. The deep reinforcement learning model, namely the deep Q-network, is exploited in this paper,

taking advantage of recent advancements in deep neural network training and analysis [48].

(Kumar, 2019) A machine learning-based Delimited Anti-Jamming system for vehicle traffic conditions that is aimed toward efficiency. The research is focused on jamming cars, and it employs differentiated signal recognition and filtering to disclose the specific position of jamming-affected vehicles. For the purpose of examining frequency fluctuations induced by signal strength changes caused by jamming or other external threats, a foster rationalizer is utilized. A machine learning open-source method was used to forecast the locations of the jamming vehicle. The approach was based on a decision tree that was based on an algorithm to anticipate the locations of the jamming vehicle. The suggested anti-jammer strategy is tested in comparison to current state-of-the-art approaches to determine its overall performance [49].

(Wu, 2019) the author made a suggestion that the increase in the complexity and connectedness of contemporary vehicles has resulted in a significant increase in the security vulnerabilities associated with in-vehicle networks (IVNs). In spite of this, current IVN designs (for example, the controller area network) do not take cyber security into mind. This problem is addressed by intrusion detection, which is a very effective means of guarding against cyber-attacks on IVNs while also guaranteeing functional safety and real-time communication assurances. As a result, it has become necessary to do more investigation. Finally, the current trend, outstanding concerns, and upcoming research avenues are discussed in further detail [50].

(Yee et al, 2019) When operating in a single perspective, many autonomous vehicle systems are unable to take use of extra scene information obtained from the viewpoint of other cars on the road utilizing vehicle-to-vehicle communication technology, which is a significant drawback. The author investigates how enhanced data sharing might improve the perceptual capacities of autonomous cars in his paper. Using a 3D sensor fusion approach, the methodology distributes sensor data and objects recognized by state-of-the-art deep learning networks amongst cars in order to boost the confidence of autonomous driving systems in identifying objects. An advantage of this strategy is that it may be used

in cases when an item is obscured (totally or partly) or positioned too far away to be adequately classified by a single-vehicle [51].

(Souhir, 2019) When communicating between vehicles (V2V), there is a difficulty with RRM. The authors used to propose a dynamic model using the Q-learning technique in which the algorithm is based on the sharing of resources and other processes of resource allocation introduces. The results of the simulations show the suggested technique provides a solution that is viable and fit while successfully improving the network performance. The authors further used the device-to-device technology which is a new technology that offers numerous benefits, including improved spectral efficiency. D2D communication is becoming more popular, it is being applied in a variety of domains such as privacy, and managing the network traffic for unloading, among others [52].

(Huang, 2019) Machine learning has lately received a great deal of interest for a wide range of applications, mostly because it works superior in managing the issues related to identification and recognition. There is a crucial use is the study using the channels, which is described below. These are the problems and possibilities associated with using algorithms for the assessment, which are discussed further in this article. A number of cutting-edge constraints, including the detection of channel line-of-sight circumstances, and the MPCs, are discussed in this article. The information collected via these procedures is used to develop accurate channel models, among other things. Furthermore, several difficulties associated with machine-learning-based data processing for the V2V channel are discussed [53].

(Gyawali, 2019) Automobile networks are vulnerable to a wide range of assaults, including denial-of-service (DoS) attacks, Sybil attacks, and false alarm generation attacks, among others. There have been a variety of cryptographic approaches suggested to defend vehicle networks from these types of cyber attacks. However, it has been shown that cryptographic approaches are less successful when it comes to protecting against insider assaults that originate inside the vehicle network system. It has been discovered that a misbehaviour detection system is more successful in detecting and preventing insider assaults. Machine learning-based misbehaviour detection system that

is trained on datasets created via intensive simulation in a realistic vehicular network environment [54].

(Kim et al., 2020) employed multi-criteria-based decision-making to overcome the limitations of the single metric routing protocols. This involved extending the existing AODV and OLSR protocols. In the AODV protocol strategies are applied to modify root request and reply mechanisms, while in the case of OLSR, a multi-path relay selection was proposed. The simulation study showed that a 15% overhead reduction was observed with a 21% lower end-to-end delay in comparison to other geographical routing protocols. However, the work badly suffered from the limited number of nodes (40 nodes) being deployed in a large network size of 6000m<sup>2</sup> which was not enough to justify the effectiveness of the approach [55].

(Patel and Ajmeri, 2020) Vehicular networks are on their way to being the most widely used and applied of all mobile ad hoc networks by the end of this century. Vehicular ad-hoc networking (VAN) is a new technology that will be used for on-the-road communications in the future. Automobile networks were expected to enable a wide range of communication applications due to the virtues of vehicle-to-vehicle and vehicle-to-infrastructure communications. These applications were expected to include diverse on-road applications in addition to deploying various road safety services. A powerful method of artificial intelligence, machine learning, may give a rich set of tools for networks to use to exploit such data for their own gain if done correctly. Specifically, in this research, the simulation will be carried out on the Road Traffic Simulator in order to offer secure communication between cars while avoiding traffic congestion [56].

(Ullah, 2020) The primary goal of this research is to examine how artificial intelligence (AI), machine learning (ML), and deep reinforcement-learning (DRL) are being used to accelerate the development of smart cities. The strategies described above are effective in the construction of optimum policies for a variety of complicated challenges relating to smart cities and urban planning. Specifically, they discussed a number of current and future research issues and areas in which the aforementioned methodologies may play a significant role in the realization of the notion of a smart city [57].

(Turan, 2021) It is the goal of heterogeneous vehicular communications to enhance security, and reliability, and reduce the delay of vehicle-based communications by incorporating numerous different communication technologies into a single communication system. Conventional fitting-based models for route loss prediction and rule-based models for radio frequency (RF) jamming detection of various communication systems both fall short of addressing the full range of mobility and jamming situations. Technologies such as Vehicular Visible Light Communications (V-VLC) provide dependable V2V communications. With the use of measurement data, the authors proposed research to employ Random Forest regression and classifier-based methods, in which numerous learners with diversity are trained and the outcome is computed by averaging the outputs of all learners [58].

(Khatri, 2021) Intelligent Transportation Systems are primarily concerned with reducing communication delays between cars and remote sensing units (RSUs), ensuring smooth traffic flow, and ensuring road safety. The VANETs have attracted the interest of several academic groups because of their use in such situations. These systems need continual monitoring to ensure correct operation, opening the door to the use of Machine Learning techniques on the large amounts of data created by various VANET applications (for example, crowd sourcing, pollution control, environment monitoring, etc.). Machine Learning is a technique in which a machine automatically learns and improves itself depending on data that has already been processed. Safety, communication, and traffic-related difficulties, as well as the infeasibility of implementing VANET systems, were discussed in detail by the author, who also studied how machine learning algorithms may be used to solve these challenges [29].

(Kang Kim et al, 2021) A neuro evolution of augmenting topologies-based adaptive beam forming technique was presented by the author to manage the radiation pattern of an antenna array and, as a result, limit the impacts induced by shadowing in urban V2V communication at intersection situations. Additional factors such as the sizes of adjacent cars and weather conditions may have an impact on communication. Communication in urban V2V communication settings becomes incredibly challenging as a result of this. Because the failure to report vehicles or accidents in time may result in the loss of human

lives, this study focuses on enhancing urban vehicle-to-vehicle (V2V) communications at junctions by using a transmission method that is capable of adjusting to the surrounding environment [59].

(Wong, 2021) For real-time on board traffic forecasts, vehicle-to-vehicle (V2V) communication is used in conjunction with satellite navigation. A hybrid strategy is presented, in which physics-based models are complemented with deep learning techniques, rather than deep learning alone. It is possible to increase the prediction accuracy by using a recurrent neural network. The hybrid model developed by the author is capable of predicting velocity up to 40 seconds in the future with higher accuracy than physics-based baselines in the field. Detailed research is being done to examine the many approaches to merging physics with deep learning that are currently available [60].

(Presses, 2021) Vehicle-to-Vehicle Communication (V2V) has made transportation safer and more efficient by enabling wireless communication among moving automobiles. The V2V physical and medium access control (MAC) layers are specified by this technology, which is the primary enabler. V2V networks are built on an ad hoc basis from vehicular stations that depend on the delivery of broadcast transmissions in order to provide the services and applications that they are envisioned to provide. When used in dense topologies, such as those seen in metropolitan networks, an appropriate MAC must be able to provide sufficient capacity for V2V communications [61].

(Bahramnejad, 2022) In the transportation business, vehicular ad-hoc networks (VANETs) are a developing technology. In order to assure road safety, vehicle communications must be reliable. In this study, an analytical approach for estimating the dependability of Vehicle-to-Vehicle (V2V) communications in Cognitive Radio (CR)-VANETs is described. The proposed framework takes into account node reliability as well as various physical, MAC, and network layer challenges for communications reliability, such as the likelihood of channel availability for CR-enabled vehicles, channel fading, transmitting vehicle contention, hidden terminal problem, and transmission redundancy. Additionally, the suggested analytical framework is used to construct a



dataset, and the reliability estimate of V2V communications is automated using an Artificial Neural Network (ANN) model [62].

(Zheng, 2022) The author studies the use of full-duplex technology in vehicular networks and develops a unique decentralized resource allocation system for full-duplex vehicle-to-vehicle (V2V) communications that is based on deep reinforcement learning and based on deep reinforcement learning. An agent is viewed as everything that exists outside of a specific V2V connection, and everything else is treated as the environment, in accordance with the mechanism. Each agent may autonomously learn about its surroundings and satisfy the needs of V2V delay limitations, all while reducing interference in the V2I communication procedure. According to the simulation findings, the total capacity of the full-duplex is greater than that of the half-duplex in terms of data transmission [63].

## 2.2 V2V Communication Based on Machine Learning

(Aznar Poveda et al, 2021) proposed an analytical model that balances transmission power and data rate in a non-cooperative network. The authors specifically train a Deep Neural Network (DNN) to exactly optimize both parameters for each vehicle without relying on extra data from neighbours or deploying any new roadside infrastructure. The results show that the proposed method not only reduces congestion but also offers adequate transmission power to satisfy the application layer needs at a given coverage distance. Last, the proposed method is thoroughly tested and assessed in three actual scenarios and under various channel circumstances, confirming its durability and superior performance in contrast to alternative solutions. The outcome in terms of PDR is 0.9, the number of decoded packets is 0.7, and the transmitter power is above 10 dBm [90].

(Choi et al, 2021) used the random forest (RF) algorithm and the long short-term memory (LSTM) method to develop a trajectory prediction method and architecture for encoders and decoders in this study. The target vehicle's row and column will be calculated using the RF algorithm and the LSTM encoder-decoder architecture, respectively, after an occupancy grid map is first created for the area surrounding the target vehicle. The test vehicle was outfitted with a camera, LIDAR sensors, and vehicular wireless communication devices for the gathering of training data, and the experiments were

carried out in a variety of driving situations. The results of the vehicle tests show that the suggested method offers more reliable trajectory prediction when compared to current trajectory prediction techniques [91].

Chbib et al, 2022 proposed an energy efficient two algorithms to secure the routing protocols against a variety of threats that aim to compromise secrecy, authentication, security, and integrity in a vehicle-to-vehicle scenario. By computing the percentage of updated destination addresses, the first algorithm determines whether each vehicle is engaging in malicious activities. This vehicle is labelled as harmful if it exceeds a set threshold. Other than that, it is a typical vehicle. The second method monitors the Signal to Interference Ratio (SIR) value, modifies the distance, modifies the power received, and modifies the sent power value in order to identify fraudulent adjustments. The authors computed the end-to-end delay, PDR, and overhead. The PDR for the Dest-anti attack is 65% while overhead is 15 for 120 vehicles [92].

(Kaur and Kakkar 2022), this research develops an attack-focused hybrid optimization-based Deep Maxout Network (DMN) in VANET. A hybrid network is used for the Cluster Head (CH) selection and routing procedure by optimizing software. To carry out an efficient classification procedure, the feature selection method is very important. Additionally, DMN is used to carry out attack classification, and it is demonstrated by optimizing software. The precision and recall of the proposed technique are 0.9395 and 0.9462 respectively while routing performance with energy and trust is about 0.25J, and 0.402 [93].

(Paranjothi and Atiquzzaman 2022) presented a framework using the statistical technique to detect rogue nodes for vehicular scenarios. The authors used the on board unit in conjunction with the fog computing model to analyse the information from different vehicles. The authors further presented an extensive simulation to create a dynamic layer and presented a network model, traffic flow model, and attack model. The outcomes in terms of TPR, FPR, throughput, overhead, and data processing time had been computed. The average throughput is 8.987Mbps while the data processing time is about 2.34ms [94].

(Cárdenas et al. 2022) presented a gradient decision tree-based routing protocol for VANET. The authors present the route map and build the model in three different steps considering the urban VANET scenario. The authors consider the different metrics such as Euclidean Distance, Trajectory Metric, Vehicle density, available bandwidth, and losses in the MAC layer. The validation accuracy is almost 100% for 20 to 50 tree depth in the training phase and 0.87 during the testing phase. The performance metrics in terms of TPR, FPR, kappa coefficient, and F-score was computed. The cohen's Kappa is 0.735 while F-score is 0.845. Further, the authors compute the packet loss ratio which is more than 20% for 150 vehicles/km, and end-to-end delay is more than 20 msec for the same. The study is limited to providing the desired results due to an increase in delay [95].

(Türkoğlu et al, 2022) has on hand an extensive catalogue of machine learning base classifiers in this regard to identify DDoS attacks directed at SD-VANETs. Primarily, a dataset with characteristics of both regular network traffic and DDoS network traffic was acquired using an SD-VANET topology that was developed experimentally. Then, the dataset's most distinctive characteristics were chosen using the Minimum Redundancy Maximum Relevance (MRMR), a selection algorithm for feature selection. Additionally, during the learning phase, the Bayesian optimization method was used to optimize the classifiers' hyperparameter. According to the experimental findings, the Bayesian optimization-based decision tree classifier and MRMR feature selection produced a 99.35% accuracy score [96].

### 2.3 Survey of Existing Routing Protocols

The most difficult task for the routing protocol is to decide the best route that is both adaptable and safe. Basically, a routing protocol is defined as a rule set that assists the packets to move from the source towards the destination within the network. There are lots of routing protocols that are used in VANET and applied according to the circumstances. Figure 2.1 shows the most widely used protocols.

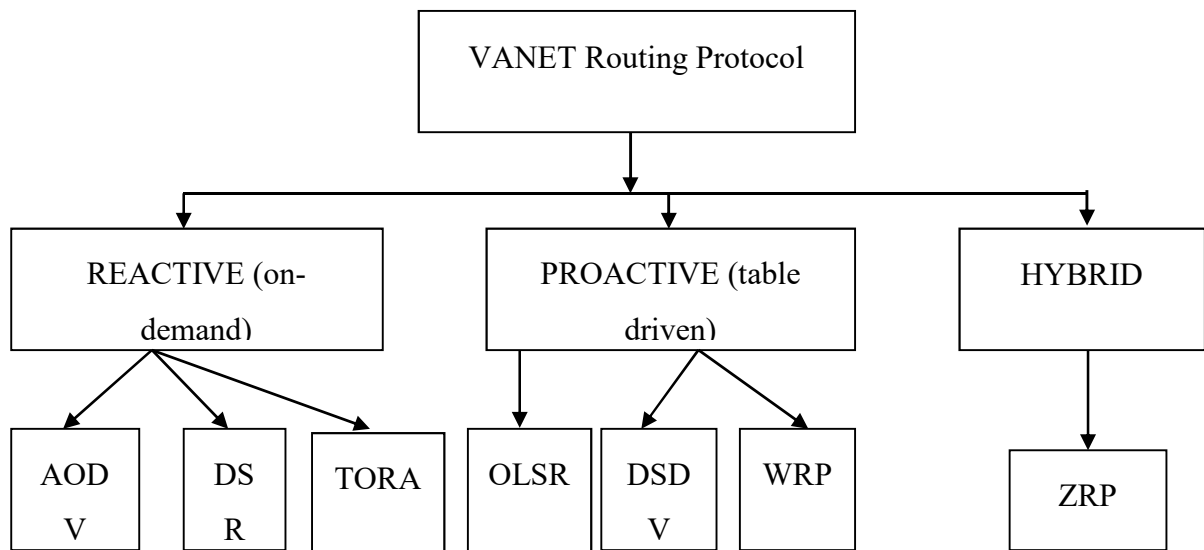


Figure 2.1 Routing Protocols [97]

### 2.3.1 Reactive (On-Demand) Protocols

The term “on-demand” says that the route is built only when it is required by the source node. It keeps these routes open for as long as the sources demand. Knowledge does not need to be disseminated in reactive protocols. When data is transmitted from the source to the destination, it absorbs bandwidth. The AODV (ad-hoc on-demand distance vector), DSR (distance vector routing), and ABR (Associatively Based Routing) protocols are all reactive protocols. On request, this routing protocol needs a route determination process. A kind of global search procedure is used if a route is needed [98].

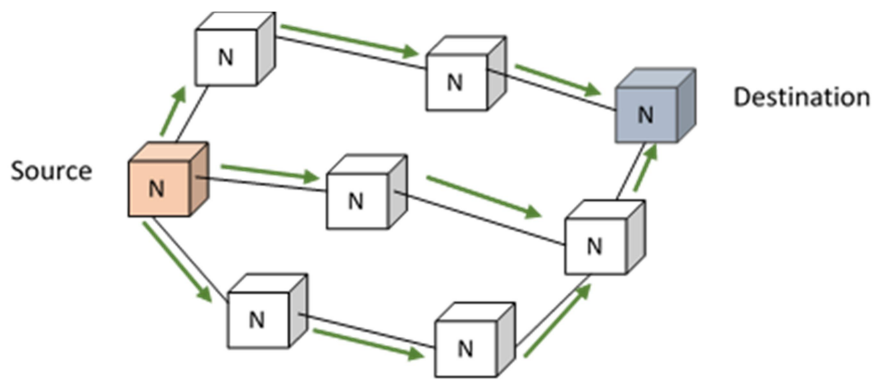
#### 2.3.1.1 Ad-hoc On-Demand Distance Vector (AODV) Routing Protocol

The communication protocols in wireless systems are developing. The protocol, named AODV as on demand routing protocol/ reactive protocol as a union of DSDV as well as DSR. A route in AODV is calculated in the same way as DSR through the route discovery process [92]. The routing table is maintained by AODV for one entry per destination, but DSR has a number of entries of route cache for every destination. [99].

The control messages in AODV are mentioned below:

- RREQ (Route Request)
- RREP (Route Reply)
- RRER (Route Error Message)

The intermediate nodes register their route tables at the address of the neighbour, from whom the first copy of the broadcast package was obtained during the RREQ pass, thus creating a reverse direction. These packages would be discarded if more copies of the same RREQ are obtained later [100]. When RREQ arrives at a destination or intermediate node with a sufficiently fresh path, the destination or intermediate node responds by uploading a route response packet (RREP) to the neighbours from whom it got the RREQ as shown in Figure 2.2.



Propagation of PREQ

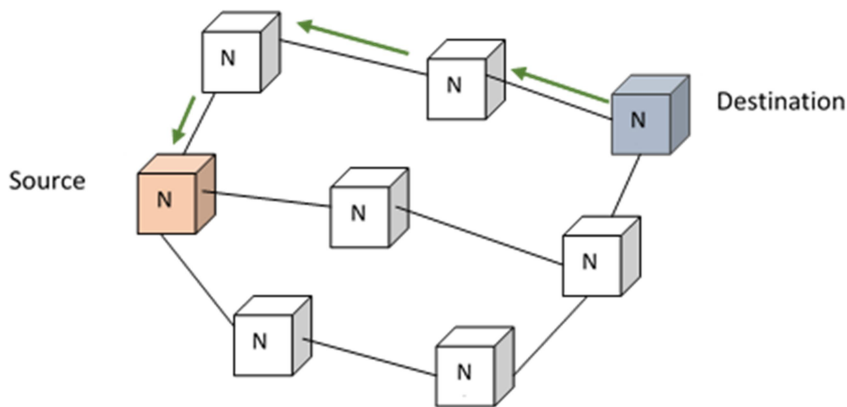


Figure 2.2 The Discovery Process

By reversing the route of RREP along with the back path, nodes in this path have set up their forward path entry into the route table pointing to the node through which the RREP arrives. The route entry also deletes any routes that haven't been used in the given amount of time. AODV only supports the utilization of symmetric links since RREP is passed on the route set by RREQ.

#### 2.3.1.2 Dynamic Source Routing (DSR)

The initial node generates Route Request (RREQ) that is forwarded over the data packets that further specify the destination as well as the source node. The salient feature of DSR is the utilization of the concept of source routing. In the source routing technique, the sender node from where the data packet is sent computes the complete sequence of nodes by which the data packet should be transferred to the destination node. That detects each forwarding “hop” through the address of the consequence node by which transmission of the packet takes place [101].

#### 2.3.1.3 Temporarily Ordered Routing Algorithm (TORA)

It is adaptive in nature, distributed routing algorithm, and is loop-free on the basis of the link exchange concept. It works with an approach that is based on source adaption to deliver multipath and loop-free routing. In this algorithm, the link reversal plays a very significant role as it controls the message flow within the formulated network and segregates them according to the topology. It mainly functions dynamically in three phases, first, it creates a route, then it maintains the created route for the data transmission and then finally it erases the route. It regularly communicates with the nodes and also assures that problems do not arise to infinity.

### 2.3.2 Proactive Protocols

It is a table-driven protocol that regularly accepts and is up to date for the series of nodes in the route. Proactive protocols attempt to determine the route in a continuous manner inside the network by which while packet forwarding requires, the route is already known so that it can be consumed immediately. The working of the protocol is maintained by using a table to take care of the current nodes inside the network. The

advantage of reactive routing protocol is that once a route is required there is little amount of delay until the route is analysed.

### 2.3.2.1 Optimized Link State Routing Protocol (OLSR)

Over the last few years, the OLSR has grown in popularity as one of the most popular proactive routing schemes. For mobile ad-hoc networks, the OLSR is developed. It is a constructive table-driven protocol, which ensures that topology information is shared with other network nodes on a regular basis. The optimization achieved with Multipoint Relay (MPRs) works well in this sense, making it well suited for large and dense mobile networks. In comparison to the classic switching state algorithm, the larger and denser a network is, the more optimization can be accomplished. OLSR uses hop-by-hop routing, which means that each node routes packets based on its local information. OLSR is better suited for networks where traffic is intermittent and random through a larger number of nodes, rather than almost exclusively between a small set of nodes. OLSR is also ideal as a proactive protocol in situations where contact pairs change over time: in this case, no additional control traffic is created because all known destinations' routes are maintained at all times. Only selected nodes, known as MPRs, are used to retransmit control messages in OLSR, which decreases the overhead stream of control traffic. The number of retransmissions needed to transmit a message to all nodes in the network is greatly reduced using this technique. Each node in the network chooses a group of nodes in its symmetric 1-hop environment from which the messages are sent. The MPR, or set of the node, is a list of selected neighbour nodes [102].

As one of the most popular proactive routing schemes, the Optimized Link State Routing protocol (OLSR) has been attracting more and more attention in recent years. The Optimized Link State Routing Protocol (OLSR) is developed for mobile ad hoc networks. It operates as a table-driven, proactive protocol that exchanges topology information with other nodes of the network regularly. It minimizes the overhead from flooding of control traffic by using only selected nodes. This technique significantly reduces the number of retransmissions required to flood a message to all nodes in the

network. Each node in the network selects a set of nodes in its symmetric 1-hop neighbourhood which may retransmit its messages.

#### 2.3.2.2 DSDV

The concept behind this routing protocol is based on the classical Bellman-Ford Routing Algorithm with a few enhancements like making it loop-free. The said routing strategy is less robust as compared to the link state routing mechanism. The two major reasons behind the same are the effect of bouncing and counting to infinity.

Under this routing protocol, the information is disseminated at a particular time interval which is the same with every node present inside the network for maintaining the changes within the routing table [103].

#### 2.3.2.3 WRP

Murthy and Garcia-Luna-Aceves [105] introduced the wireless routing protocol. The main goal is to keep track of the shortest distance between each destination and all nodes in the network. WRP stands for Wireless Routing Protocols and is known as a loop-free routing protocol. WRP is a path-finding algorithm that avoids the count-to-infinity problem by requiring each node to verify the accuracy of predecessor information recorded by all of its neighbours. To keep more accurate details, a set of four tables are used by each node within the network. These are the Message Return Table (MRL), Call Cost Table (LCT), Routing Table (RT), and Distance Table (DT) [104].

#### 2.3.3 Hybrid

This type of protocol is formed by overcoming the disadvantages of both proactive and reactive protocols. Reactive protocol is economic whereas proactive protocol's latency rate is small. Therefore by using these two advantages, a hybrid routing protocol is developed. This protocol is a combination of proactive as well as reactive routing protocols. It uses a mechanism that requires a jet protocol and a proactive protocol table mechanism to eliminate overlap network problems. The hybrid protocol is best suitable for large size of networks. In addition, routing in a large region of the



network is done outside of the proactive approach and the zone routing is done through a reactive approach and divided into several zones among them Zone Routing Protocol (ZRP) is the most important routing protocol [104].

Out of the studied routing protocols made in the literature survey [38, 42, 51, 53, 55, 66], AODV has been studied as one the most suitable routing architectures to be implemented in VANET. A detailed comparative analysis of the existing studies for addressing various types of attacks in VANETs is summarized in table 2.1. The table also highlights the machine learning techniques used by the authors with the possible limitations of their studies.

Table 2.1 Comparative Analysis of Approaches for Addressing Attacks in VANETs

| Authors and citation          | Technique       | Attack                    | Dataset                            | Findings  | Limitations  |
|-------------------------------|-----------------|---------------------------|------------------------------------|---|--|
| Kang, I., (2012) [106]        | SVM             | DoS, U2L<br>R2L,<br>Probe | DARPA 1999                         | Higher capabilities in terms of accuracy for attack detection               | Suitable for specific types of attacks and face difficulty in the detection of unknown attacks |
| Hu, W., (2014). [107]         | GMM+<br>PSO+SVM | DoS, U2L<br>R2L,<br>Probe | KDD Cup 1999                       | Higher accuracy with lower FAR during anomaly detection                     | Models consume more time in computation  |
| Alom, M. Z.,(2015) [108]      | DL              | DoS, U2L<br>R2L,<br>Probe | NSL-KDD                            | The model has a higher rate of attack detection and classification accuracy | In complex scenarios model becomes expensive   |
| Leandros, A., M. (2015) [109] | SVM+KNN         | DoS,                      | Open-source real-time traffic flow | Higher anomaly detection  | Higher FAR in dense traffic flow   |

|                                |   |                      | dataset                                   | accuracy and lower computation time  | scenarios  |
|--------------------------------|---|----------------------|---|--|--|
| Aburomman, A. A., (2016) [110] | SVM + KNN + PSO                                 | DoS, Probe, R2L, U2L | KDD Cup 99,                               | Higher accuracy of attack classification   | High computation cost  |
| Jabbar, M. A., (2017) [111]    | DT  | DoS, U2L R2L, Probe  | Kyoto                                     | Higher accuracy with a minimum error rate  | The model has Low computation speed in a heavy traffic scenario        |
| Shone, N., (2018) [112]        | DL  | DoS, U2L R2L, Probe  | KDDCup99 and NSL-KDD                      | the proposed model shows promising results in terms of accuracy and time reduction | The model is weak in real-world zero-day attacks                       |
| Liang, J., (2019) [113]        | Feature extraction and I-GHSOM-based classifier | Sybil attacks        | Open-source vehicle BSM and map road data | Higher attack detection accuracy, processing efficiency                            | Performance of the proposed model affected in a dense traffic scenario |
| Boukerche, A., (2020) [114]    | DL model using GCN                              | DoS                  | Real-time 39000 individuals dataset       | Higher accuracy in anomaly prediction  | Model is expensive in practical scenarios                              |
| Batchu, R. K., (2021) [115]    | LR+DT +GB+ KNN and SVM.                         | Distributed DoS      | CICDDoS2019 dataset                       | High classification accuracy upto 99.97%   | In practical scenarios, the model consumes time for computation        |
| Türkoğlu, M., (2022) [116]     | Bayesian optimization-based DT classifier       | Distributed DoS      | 42-feature dataset                        | Higher accuracy upto 99.35 % for attack detection in VANET                         | Higher computation time  |

|                        |                                   |                        |   |   |  |
|------------------------|-----------------------------------|------------------------|---|---|--|
| Chen, Y. (2023), [117] | MDFD framework with ML classifier | Sybil attacks with Dos | Open-source vehicle BSM data, map road data, and road detector feedback dataset | Higher average attack detection accuracy upto 97.69% in VANET | Detection performance is affected in the case of multi-variable dense traffic flow |
|------------------------|-----------------------------------|------------------------|---|---|--|

GMM: “Gaussian mixture model” FAR: “False alarm ratio”, DL: “Deep learning”, LR: “Logistic regression”, DT: “Decision tree”, GB: “Gradient boost”, MDFD: “A multi-source data fusion detection”, I-GHSOM: “improved growing hierarchical self-organizing map”, GCN: “Graph Convolutional Network”

## 2.4 Research Gaps

The architecture of the proposed work is dependent upon the establishment of trust via machine learning among the vehicular nodes to attain maximum QoS values. The presented literature demonstrates various types of models and methods that utilize machine learning for trust generation and evaluation. Machine learning-oriented trust generation requires training in bulk amounts and hence correct labelling of the data against defined trust is also an important step to be observed. The studied literature has inspired the proposed work to quite a good extent but there is always a scope for improvement. Based on the studied literature, the following gaps have been identified.

- A) Machine learning-based training and classification architecture have been proposed for trust generation using the Random Forest regression model but no preliminary analysis of the best suitable routes for each category has been done [58].
- B) Neuro-based learning mechanism has been presented for V2V communication prevention that utilized deep learning straight forward. The regression model that is opted for by [58] could have been a part to improve the training that would have eventually reduced the prediction time and also would have increased the QoS [60].
- C) Machine learning is oriented towards better training and classification mechanism. In order to train the system, data aggregation is one of the most

preliminary steps to be processed. The authors aggregated the dataset through a simulation environment and passed it to a neural network but did not disclose the labelling architecture of the attacks [62].

- D) The authors proposed a machine learning-oriented anti-jamming system and also utilized multi-input and multi-objective training and classification modules. The proposed work by the author is only limited to jamming and the authors do not consider the possibility of jamming due to security attacks [49].

## 2.5 Summary of the Chapter

This chapter highlights the existing research work on VANET using the different routing protocols. A survey has been presented based on Machine learning, DDoS attacks, threats, and literature represented in tabular form to describe the techniques and the type of attacks addressed by the researcher in the recent past. Based on this review the next chapter will present the research methodology involved in the proposed work.

## Chapter 3: Research Methodology

---

The chapter is dedicated to presenting an overview of the research methodology that is followed in the proposed work to improve the security of V2V communication in VANETs. The process takes advantage of the machine learning algorithm to learn and define trust level.

### 3.1 Research Methodology

VANET involves communication architecture that involves routing strategies and communication protocols that exists between multiple vehicles in the network. In order to communicate the data from source to destination, it becomes necessary to add intermediate hops to prevent loss in time and packets as the terminal might be quite far from the source. When VANET is concerned, the vehicles are allowed to roam freely in the network, and in such a scenario, it becomes vital to choose the nodes that can be relied upon for communication. Taking advantage of the mobility in the network, malicious nodes may also affect communication. Holding the communication for a long interval to analyse the path every time, will increase the computation complexity of the network as well. The labelling of the nodes has been done on the base of QoS parameters. A node that performs consistently well in all the contributed routes viz. the routes in which the node is participating, results in higher throughput and less delay. Even if a single node is affected by any physical failure or network failure in the route, it will result in degraded QoS. The overall network performance has been computed by calculating the normalized average of the parameters and hence, a node can be labelled as a trusted node, which results in secure route formation, when the QoS parameters involving the node attain more than a specific threshold of network QoS. For the proposed work scenario, four QoS parameters have been considered namely Throughput, Packet Delivery Ratio(PDR), Jitter, and True Detection Rate(TDR) and they can be mathematically explained as follows.

- a) Throughput= It is the total received packets per second and can be mathematically defined as follows

$$\text{Throughput} = \frac{R_p}{t} \quad (3.1)$$

Where  $R_p$  is the total received packets in a 't' time interval.

- b) PDR: It is the ratio of the delivered packets to the sent packets and mathematically it can be presented as follows.

$$\text{PDR} = R_p/S_p \quad (3.2)$$

where  $S_p$  is the sent packets.

- c) Jitter: It is the total delay in the communication for the discovered routes. Mathematically, it can be defined as follows.

$$\text{Jitter} = \sum_{i=1}^p d \quad (3.3)$$

where  $d$  is the route delay

- d) TDR: It is the True Detection Rate of the affected node and mathematically it can be defined as follows.

$$\text{TDR} = \int_{t=1}^{t'} \text{Number of Successful}_{\text{packet transmission}} \quad (3.4)$$

A communication network is dependent upon a lot of factors and it is not necessary that the power consumption rises only due to the issues in the route. A buffer jam in the communication buffer of the VANET node may also increase power consumption. This dissertation views the VANET as a node architecture of  $n \times m$  where  $n$  is the total number of nodes in the network and 'm' is the total number of QoS parameters that are evaluated with the establishment of the route.

The problem of this research work is to segregate the communication route into multiple class labels to signify the trust value of the network. A high trust worthy node will be preferred when the route is to be established. Machine learning architectures have put a significant mark in the world for the efficacy of the modern world. This research work also aims to dignify the usage of machine learning for the improvement of the QoS parameters of communication. VANET architecture may involve multiple types of communication systems and multiple types of intrusion in the network. DoS is one of the most common attacks that have been identified in VANETs. There are similar type of

attacks that follows the pattern of DDoS attack with alterations in the threat architecture and hence the problem formulation extends to the development of a generic solution. Proposed work aims to design a generic solution for the prevention of DDoS and similar kind of security threats. A generic solution refers to an algorithmic solution that can identify multiple attacks that possesses a similar kind of nature as that of a DDoS attack. For example, the proposed work can also be utilized on replay attacks wherein the ground truth values have to be marked with the aggregated data against replay attacks. As any novelty requires significant proof of improvement, the problem statement extends to compare the proposed work with another state of art algorithms and techniques.

### 3.2 Objectives

Below are the overall objectives of the proposed work which are derived on the basis of a thorough literature survey.

- To find out an improved methodology to differentiate between trusted and untrusted nodes.
- To propose a packet efficient prevention architecture to enhance the secure V2V communication based on Machine Learning
- To propose an architecture which is not restricted to only DoS related attacks
- To evaluate and compare the proposed architecture with previous prevention architectures based on Quality of Service (QoS) like Jitter, PDR, TDR and Throughput.

### 3.3 Graphical Representation of the Work

To achieve the above-listed objectives a graphical representation of the whole methodology is presented in figure 3.1. In the initial stages, the network is deployed by defining the network nodes, network area, road side units, source, destination, etc. in MATLAB. The cluster head is defined to pursue with the communication using AODV protocol and dragon fly based optimization. This is followed by the identification of the trust worthy nodes among the available nodes to form clusters. These clusters are again

labelled based on a fuzzy rule set followed by the multilayer neural architecture. The performance of this trust worthy V2V communication is evaluated in terms of throughput, PDR, TDR, and jitter. All the steps of this research are shared in the form of a flowchart and the implementation of the same is elaborated in the upcoming chapters.



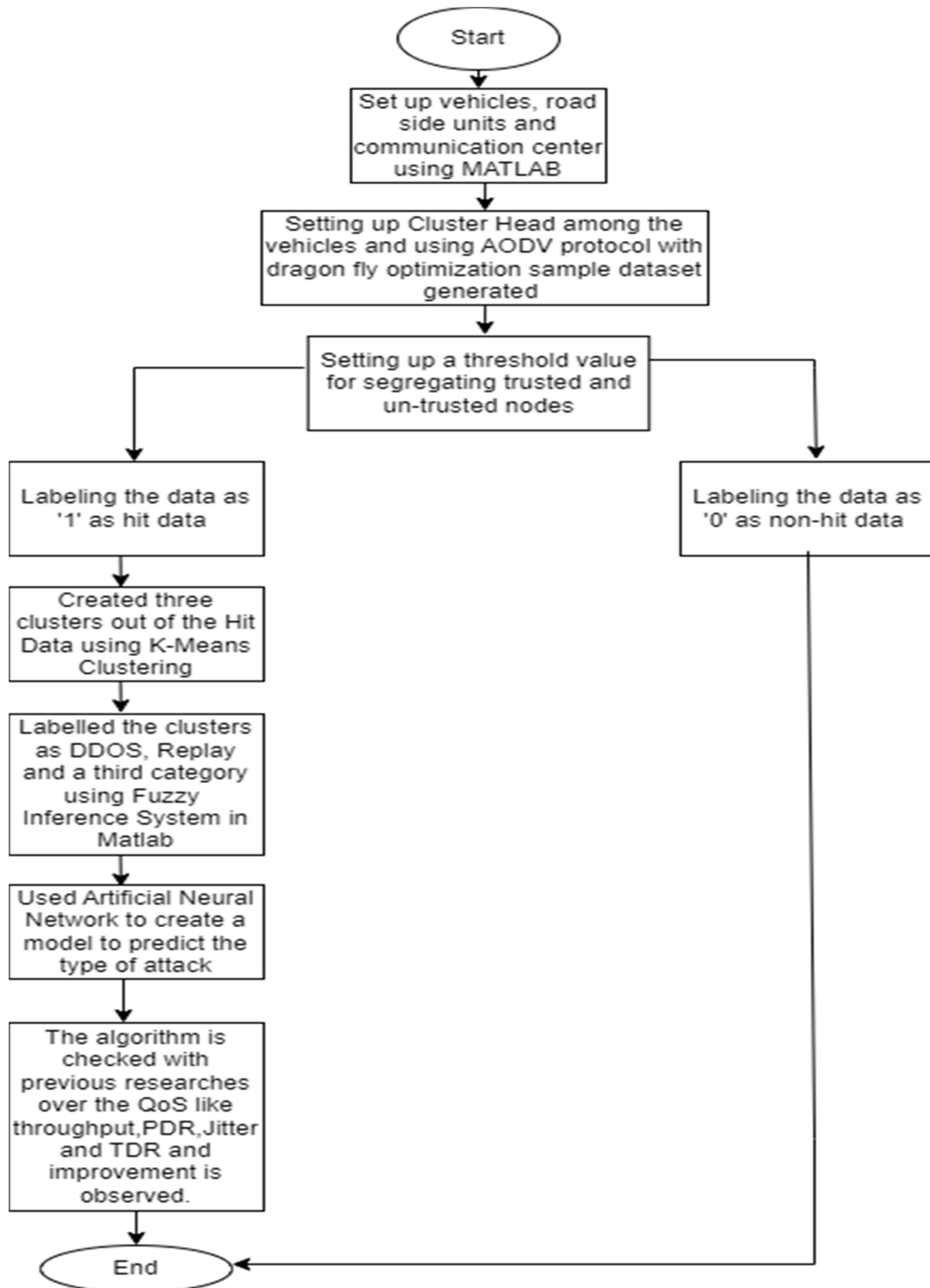


Figure 3.1 Graphical Abstract of the Proposed Work

### 3.4 Summary of the Chapter

The present chapter discusses the problem and challenges that need to be addressed. Based on the problem statement the objectives designed for the present research are listed in the chapter. A graphical representation of the implementation is shared which simplifies the understanding of the work done in this research.

## Chapter 4: Trust Identification

---

### 4.1 Introduction

A node or vehicle is said to be trustworthy in V2V communication if the node transfers the desired data to the desired terminal in provided time slab. V2V communication refers to the establishment of the channel between one user vehicle and another user vehicle. The vehicles share information in order to make communication more convenient and fruitful. A V2V communication will contain the following set of objects or elements.

- a) Vehicle
- b) Road Side Units (RSUs)
- c) Communication Channel

The vehicles transfer the information through the communication channel. The evaluation of the channel can only be done once the data is transferred from the source vehicle to the terminal vehicle. The proposed research work has utilized MATLAB as the simulation tool for the implementation of the designed algorithmic architectures. In order to simulate a communication model, routing protocols have been implemented in MATLAB.

### 4.2 Implementation Design

The AODV protocol has been designed and developed on MATLAB. The implementation design is illustrated as follows.

- a) Node deployment and simulation setup

An increasing set of communication nodes have been designed to be used in the implementation behaviour. As the AODV routing protocol, broadcasts its requirement to the neighbouring nodes, a coverage set calculation has also been implemented. As the proposed work architecture considers two-dimensional pattern analysis, the distance between the vehicle and the communication center has been calculated using the distance formula given by eq (4.1)

$$dcc_i = \sqrt{(nx(i) - cx)^2 + (ny(i) - cy)^2} \quad (4.1)$$

where,

$nx$  = vehicle's x location

$ny$  = vehicle's y location

$cx$  = communication centre's x location

$cy$  = communication center's y location

Calculation of RSU Count: In order to propose an optimum model, following formula is used for calculating the count of required RSUs

When vehicle < 50,

$$\text{Count of RSU} = \left( \frac{\text{wid}}{\sqrt{\text{no of vehicles}}} * \sum_{i=1}^n \frac{\text{DisCC}_i}{N} \left| \sqrt{2\pi \text{spd}} * \text{Len} \right) + k \quad (4.2)$$

otherwise,

$$\text{Count of RSU} = \left( \text{wid} * \sum_{i=1}^n \frac{\text{DisCC}_i}{N} \left| \sqrt{2\pi \text{spd}} * \text{len} \right) + k + \frac{n}{n_k} \quad (4.3)$$

where

$\text{DisCC}_i$ : distance of the vehicle from the communication center

$\text{wid}$ : width of the transmission area

$\text{len}$ : length of the transmission area

In order to validate the provided formulas in (4.2) and (4.3) the following scenarios have been created and the formula is validated accordingly.

The value of k is kept constant where  $k=2$  and  $n/n_k=2$ ;

Table 4.1 RSU Formula Validation

| Deployment Area | Avg Velocity | Number of Vehicles | Avg distance from the communication center | RSU Count | Scenario |
|-----------------|--------------|--------------------|--|-----------|----------|
|-----------------|--------------|--------------------|--|-----------|----------|

|            |         |    |        |   |  |
|------------|---------|----|--------|---|--|
| 1000 -1000 | 40 km/h | 40 | 167.32 | 4 | Area=1000,<br>V=40                                       |
| 2000-2000  | 40 km/h | 80 | 190.22 | 7 | Area, Vehicles<br>increased by<br>50%                    |
| 2000-2000  | 40 km/h | 40 | 220.11 | 5 | Only Area<br>Increased,<br>Vehicles are<br>kept constant |
| 1000-1000  | 40 km/h | 80 | 123.32 | 4 | Vehicles<br>increased area<br>is constant                |

The proposed algorithm encapsulated a mechanism based on ranking in route discovery which is defined in the “algorithm route discovery” later in the same chapter. It considers that every kind of delay has a certain reason and every particular delay is just another setback to the already existing challenges mentioned in Sections I and II. Since RSUs are specifically accountable for the smooth circulation of the shared data, the considered delays are for Road Side Units.

$$fn(i) = \int_{i=1}^{RU_{count}} \log_2\{Dr_n(i) + Dr_d(i)|Dly_n(i) + Dly_d(i)\} \quad (4.4)$$

where,

$Dr_n$  = Drop in normal mode

$Dr_d$  = Drop in distortion mode

$Dly_n$  = Delay in normal mode

$Dly_d$  = Delay in distortion mode

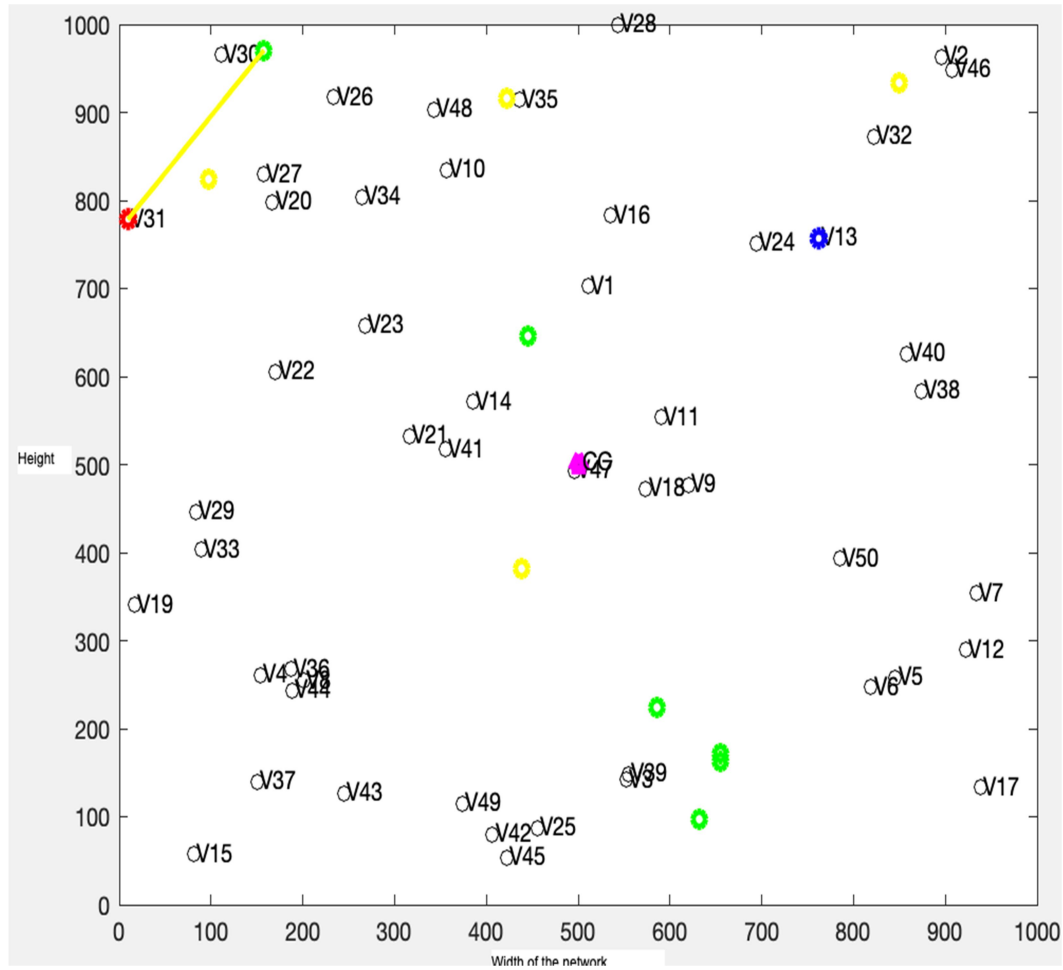


Figure 4.1 Node Deployment

To deploy the nodes, a random hybrid generation model is designed. The total network length as shown in Figure 4.1 is set to be 1000 meters which in further implementation has been varied to an extent of 3000 meters in both width and height for more simulation records. Random selection of source and random selection of destination has been made in the proposed architecture. The route discovery process will follow the broadcast mechanism in which each vehicle broadcasts its requirements to its neighbouring vehicles. If the vehicle is interested in sending responses, its response is counted as R-REP against the route request R-REQ. As trust has been counted as a primary objective of this research, a route will also contain network disruption to justify the trust factor in the network. In order to compute the Cluster Heads, the entire region is divided into C clusters. The total number of clusters depends upon the total length of the network and the total deployed nodes in the network. As the proposed algorithm is a cluster and

broadcast-supported routing architecture, the nodes are initially categorized based on the location of the vehicles. To do so, the nodes have been initiated with a 10% CH value viz. the total CH count will be 10% of the total vehicles. If the distribution density increases by 20% viz. total nodes/total deployment area increase by 20%, there will be an addition to 1 more CH. That satisfies 5 CHs background for 50 vehicles i.e. if the density increases by 100%, 5 more CHs will be deployed.

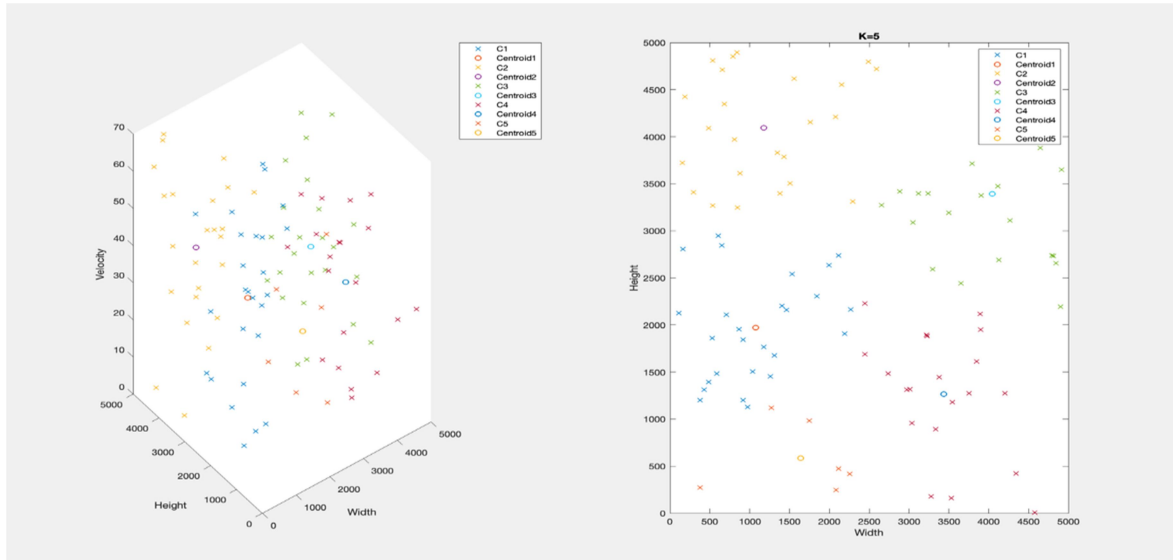


Figure 4.2 The Vehicle Distribution

The 3d portion illustrates the x, and y location of the vehicle and initial velocity of the vehicles. As the vehicles are mobile, they might enter into another cluster at the very initial instance of communication and hence the CHs are only chosen on the base of the initial velocity of the vehicle. The least velocity will represent a stable node to be suitable for communication. In order to calculate the velocity of the vehicle, the proposed work uses the last known location.

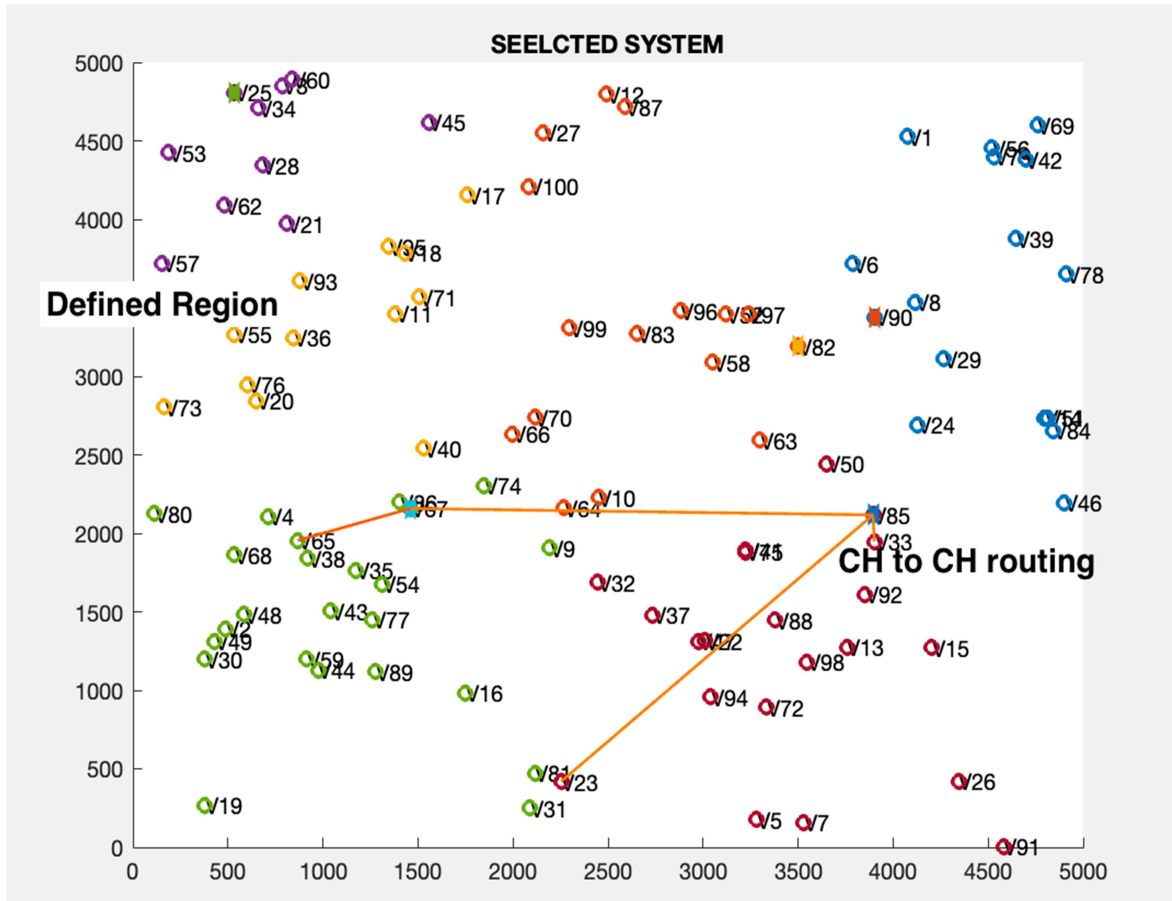


Figure 4.3 The Communication Network

In order to understand the discovery procedure, route V23-V85-V64-V65 can be observed. The procedure involves broadcast requests from all the participating nodes. As a distributed attack system is considered, multiple attackers are placed at multiple locations to disrupt the communication procedure as also observed in this route. The transfer mechanism considers the following attribute aspects illustrated in table 4.2.

Table 4.2 Simulation Architecture

|                            |                  |
|----------------------------|------------------|
| Total number of nodes      | 50 – 100         |
| Area Size                  | 1000-3000 meters |
| Sensing range              | 200 meters       |
| Communication Channel used | Rayleigh         |



|                                |   |
|--------------------------------|---|
| Packet Injection Rate (PIR)    | 10,000 packets/second   |
| Attack Models                  | DDoS, Replay  |
| Evaluation Criteria            | Throughput, Packet Delivery Ratio, Jitter , TDR   |
| Software/Hardware Requirements | <ul style="list-style-type: none"> <li>a) CPU : I3 and Above , M1 and above for MAC</li> <li>b) RAM: 4 GB and above</li> <li>c) Cache: 2MB and above</li> <li>d) Cores: 2 and above</li> <li>e) Operating System: Windows 10 and above / MAC 2.0 and above</li> </ul> |
| Simulation Tool                | MATLAB 2016 and above   |

The proposed route discovery algorithm is inspired by Yuvraj et al. and Kim et al. The base description is as follows.

Yuvraj et al. They focus on handling quality of service (QoS), congestion avoidance, and congestion aware routing in the presence of asymmetry. Asymmetry refers to the unequal channel characteristics experienced by wireless devices, such as differences in bandwidth, delay, and error rates between the uplink and downlink directions. The authors' proposed work aims to improve network performance and resource utilization in the presence of asymmetric channels. They propose a cross-layer framework that incorporates QoS provisioning, congestion avoidance, and congestion aware routing mechanisms. By leveraging information from multiple layers of the protocol stack, their approach can adaptively adjust the network parameters based on the asymmetry characteristics and traffic conditions.

The methodology employed in this work involves the following steps:

1. Characterization of Asymmetry: The authors first analyse the asymmetry characteristics in the wireless channel, including bandwidth, delay, and error rates. This step provides insights into the nature and extent of the asymmetry present in the network.

2. **Cross-Layer Framework Design:** Based on the asymmetry analysis, the authors design a cross-layer framework that integrates QoS provisioning, congestion avoidance, and congestion aware routing mechanisms. This framework enables efficient resource allocation and routing decisions, taking into account the asymmetry characteristics.
3. **Quality of Service Provisioning:** The proposed approach incorporates QoS provisioning techniques to ensure that the desired performance requirements are met for different types of traffic. QoS parameters such as delay, jitter, and throughput are considered to provide differentiated services based on application requirements.
4. **Congestion Avoidance:** The authors introduce congestion avoidance mechanisms to prevent network congestion, especially in asymmetric scenarios. These mechanisms dynamically adjust the transmission rates, buffer sizes, and congestion control parameters to regulate the flow of traffic and maintain optimal network performance.
5. **Congestion Aware Routing:** The proposed approach also addresses congestion aware routing, where routing decisions are made based on the current congestion levels in the network. By considering the asymmetry characteristics and congestion information, the routing algorithm can choose the most suitable paths for data transmission. The outcome of the research is an extended cross-layer approach that effectively handles asymmetry in wireless networks. By integrating QoS, congestion avoidance, and congestion aware routing mechanisms, the proposed approach improves network performance, enhances resource utilization, and provides better service differentiation for different traffic types. The methodology used in this work enables adaptive adjustments based on the asymmetry characteristics and current network conditions.

Kim et al. presents an extension OLSR and AODV protocols using a multi-criteria decision-making method. The authors aim to enhance the routing performance of these protocols by considering multiple criteria, such as energy efficiency, throughput, and delay, during route selection. The proposed work focuses on improving the efficiency and

effectiveness of routing decisions in ad hoc networks. Traditional routing protocols often consider only a single metric, such as hop count, to determine the best route. However, this approach may not always result in the optimal choice, especially when multiple criteria need to be taken into account.

The methodology employed in this work involves the following steps:

1. **Identification of Routing Metrics:** The authors identify multiple criteria, including energy efficiency, throughput, and delay, that are crucial for determining the performance of the routing protocols. These metrics reflect different aspects of network performance and user requirements.
2. **Multi-Criteria Decision-Making Method:** The proposed approach incorporates a multi-criteria decision-making method, such as the Analytic Hierarchy Process (AHP) or the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS). These methods enable the evaluation and comparison of different routes based on the identified metrics.
3. **Extension to OLSR and AODV:** The authors extend the OLSR and AODV routing protocols to incorporate the multi-criteria decision-making method. By considering multiple metrics, the extended protocols can select routes that optimize the overall network performance, taking into account the preferences of different users and the network conditions.
4. **Route Selection and Maintenance:** The extended protocols make use of the multi-criteria decision-making method to select routes that satisfy the desired criteria. Additionally, mechanisms are implemented to handle route maintenance, ensuring that the selected routes remain valid and efficient.

The outcome of this research is the extension of the OLSR and AODV routing protocols to incorporate a multi-criteria decision-making method. By considering multiple metrics during route selection, the extended protocols can make more informed decisions that result in improved overall network performance. The methodology used in this work enables the selection of routes that satisfy user preferences and network conditions, leading to enhanced energy efficiency, throughput, and delay in ad hoc networks.

The proposed route discovery mechanism combines the algorithmic approach of AODV combined with Swarm intelligence (SI) based dragonfly algorithm (DFA). The dragon fly algorithm was proposed in 2015 and has been utilized in route discovery policies for a long time now. The general architecture of the dragon fly algorithm comprises of the food search behaviour of the dragon flies or the behaviour that is observed when the dragon flies migrate from one region to another region. The dragon algorithm is based on the optimization policy that every group of dragon flies will have a local best solution and the flies will have to find ways to reach the global best food in order to remain updated about the best food value. A set of dragon flies is divided into multiple swarm sizes in which every dragonfly is paired with several other dragonflies in a repeated number of simulations. The paired dragon fly may get old members from any previous group or may get a completely new set of pairs. It is referred to as Random Walk Model(RDM) and is defined in terms of Levy flights. The dragon fly algorithm is evaluated on the base of three factors as follows.

- A) Cohesion: It is defined as the tendency of the dragon fly to reach the global best solution
- B) Alignment: It is defined as the change in velocity or direction of a dragon fly.
- C) Separation: It is defined as the total distance from other flies that may collide with the evaluated dragon fly. The measurement can be both inter and intra.

The proposed DFA algorithm can be illustrated using the following Algorithmic representation. The proposed DFA algorithm starts once the broadcaster or RREQ gets R-REP from the nearby CHs. Each CH checks its packet buffer that defines the total number of packets in a buffer of existing data type. The CH calculates the total idle time based on the execution rate of the sensor and returns the total idle cost. The replicant node does not send its current velocity to the broadcaster node and hence the broadcaster will have to evaluate the possible location of the CH.

---

*AlgorithmRouteDiscovery*

---

Input:  $R_{\text{HistorySheet (RHS)}}$ , Destination (Dn), Source (Sr) and Output: Optimal Route

//RSUHistory contains the history belonging to the previously occurred communication between any given RSU and a vehicle that needs to send the data. The delays listed in Eq(4.4) are calculated according to the RDM model discussed previously.//

1. Route = [];
2. Src<sub>Index</sub> = Identify the source from RHS //Identify the index of the sources in the record history sheet//
3. Dstn<sub>Index</sub> = Identify the destination // Identify the index of the destination in the record history sheet//
4. Route[1] = Src. Id
5. Link<sub>List(LIST)</sub> =  
Identify the shared links between Sr and Dn from previous communication involving RSUs
6. Calculate all the drop patterns as per Eq(4)and calculate the rating factor by using the same equation
7. Src<sub>info</sub>(Si) = RSh. Source. History();//Fetching the history of the Source
8. Dstn<sub>information</sub> (Di) = RSh. Dstn. Hstry() // Follow step 7
9. Average\_Info =  $\frac{1}{2} \times \left[ \frac{1}{t} \times (\sum_{p=1}^t \text{Sri}) + \frac{1}{t} \times (\sum_{p=1}^t \text{Dsi}) \right]$  Eq (4.5 )
10. In order to create an information list, the proposed work has utilized the Dragon Fly Algorithm(DFA), and the illustration of the route discovery is provided as follows.
11. D<sub>r</sub> = [ ]// Initialize and empty root
12. flag<sub>d</sub> = 0 // Indicates whether the destination has been found or not, 0 for no and 1 for yes
13. D<sub>r</sub>[0] = S // First node of the route is a source
14. D<sub>r</sub>[1] = CH<sub>S</sub> // Second node of the route is CH of the source
15. T<sub>S</sub> = CHs // Create a temporary source
16. while<sub>1</sub> (flag<sub>d</sub> == 0)
  - a. Response = Broadcast(T<sub>S</sub>).RREP // Start broadcast from a temporary source
17. D<sub>f</sub> = Responses // Consider each response as one Dragon fly
18. L<sub>f</sub> = D<sub>f</sub>. count // Consider 5 levy flights
19. D<sub>f</sub><sub>Result</sub> = [0, L<sub>f</sub>] // Create a matrix for the dragon result
20. For j = 1: D<sub>f</sub>

21.  $E_v = \frac{\sum_{k=1}^t \frac{D_s}{L_{c_t} - C_{c_t}}}{t}$  where  $D_s$  is the total distance of  $D_{f_j}$  from its co-ordinates measured at the time  $L_{c_t}$  and current time  $C_{c_t}$
  22.  $E_{QoS} = \frac{\sum_{k=1}^t [D_{f_j}.Throughput, D_{f_j}.PDR]}{t}$  // Extract the QoS service parameters of the old t number of communications  
The value of  $E_v$  is unknown if there is no previous communication between the current respondent dragonfly and the broadcaster and hence in such a scenario,  $E_v$  is set to be maximum
  23.  $E_{value} = |E_{QoS} - E_v|$
  24. Alignment =  $E_{value}$
  25. In case, there is no communication history, there would be no response sheet, and hence the value of  $E_v$  and  $E_{QoS}$  will be 0. In such a scenario, if each CH is unknown, they all fall into the same trust value which is referred to as alignment.
  26. Cohesion=1 if  $D_{f_j}$  holds destination in a coverage list file  $g_d = 1$
  27. Cohesion=1/ $D_{TSD_{f_j}}$  where  $D_{TSD_{f_j}}$  is the distance between the temporary source and the broadcaster
  28. Dragon<sub>fit</sub> = norm  $|\frac{Alignment}{Cohesion}|$
  29.  $D_{f_{Result}}[j]=Dragon\_fit$
  30. End For
  31.  $D_r.$  Add(Index. max  $D_{f_{Result}}$ ) // Add CH with maximum index
  32. end while<sub>1</sub>
  33. Return  $D_r$
- 

By the end of the simulation architecture, a total of 10,000 simulation results have been collected. As a result of the proposed dragon fly algorithm, a route will be discovered consisting of the routes in the network, total network hits, and QoS parameters that are represented in table 4.3. Due to its huge size, it cannot be represented entirely in this report but a sample size of 100 simulations is represented in the Table 4.3. In the drawn table, vehicle 26 has CH1 and CH1 finds CH6 using the firefly algorithm discussed in the algorithm route discovery.

1. 'Route': The 'Route' column represents the nodes visited in the order of the route. For the first route, the nodes visited are 26, 1, 6, and 36, while for the second route, the nodes visited are 18, 8, 4, and 25.
2. 'Hits': The 'Hits' column represents the number of times each node is hit. For the first route, nodes 26 and 1 are visited 9 & 10 times each, while nodes 6 and 36 are hit 0 times each. For the second route, node 18 is visited once, node 8 is hit twice and 4 & 25 are not hit at all.
3. 'RSUs': The 'RSUs' column represents the Road Side Units (RSUs) utilized by the route. For the first route, RSUs 1 and 6 are utilized, while for the second route, RSUs 8 and 4 are utilized.
4. 'Vehicles': The 'Vehicles' column represents the vehicles used for the route. For the first route, vehicles 26 and 36 are used, while for the second route, vehicles 18 and 25 are used.
5. 'Total Packets': The 'Total Packets' column represents the total number of packets transmitted during the simulation. In both routes, 6000 packets are transmitted.
6. 'Throughput': The 'Throughput' column represents the amount of data transmitted per unit of time. The first route has a throughput of 2.07892901, while the second route has a throughput of 2.06669936.
7. 'PDR': The 'PDR' column represents the Packet Delivery Ratio, which is the percentage of packets transmitted successfully. The first route has a PDR of 0.94689225, while the second route has a PDR of 0.94470792.
8. 'Jitter': The 'Jitter' column represents the delay variation between packets. The first route has a jitter of 217.988594, while the second route has a jitter of 203.457459.
9. 'TDR': The 'TDR' column represents the True Detection Rate which is the percentage successful packet transmissions made during the data transmission. The first route has a TDR of 94.13106498, while the second route has a TDR of 92.86305972.

Table 4.3 Sample Simulation Result

| 'Route' | 'Hits' | 'RSUs' | 'Vehicles' | 'Total packets' | 'Throughput' | 'PDR' | 'Jitter' | TDR |
|---------|--------|--------|------------|-----------------|--------------|-------|----------|-----|
|---------|--------|--------|------------|-----------------|--------------|-------|----------|-----|

|                 |                 |       |         |      |                |                |                |                 |
|-----------------|-----------------|-------|---------|------|----------------|----------------|----------------|-----------------|
| [26,1,6,3<br>6] | [9,10,0,0<br>]  | [1,6] | [26,36] | 6000 | 2.078929<br>01 | 0.946892<br>25 | 217.9885<br>94 | 94.13106<br>498 |
| [18,8,4,2<br>5] | [1,2,0,0]       | [8,4] | [18,25] | 6000 | 2.066699<br>36 | 0.944707<br>92 | 203.4574<br>59 | 92.86305<br>972 |
| [9,8,4,1]       | [4,0,0,0]       | [8,4] | [9,1]   | 6000 | 2.016844<br>25 | 0.922985<br>92 | 151.7308<br>78 | 95.48405<br>124 |
| [26,5,4,7<br>]  | [0,13,8,2<br>]  | [5,4] | [26,7]  | 6000 | 1.993248<br>64 | 0.913362<br>6  | 125.3856<br>04 | 95.43535<br>286 |
| [23,7,31]       | [14,9,0]        | 7     | [23,31] | 6000 | 2.019639<br>83 | 0.929071<br>37 | 122.1152<br>37 | 98.44588<br>8   |
| [21,6,4,2<br>5] | [6,0,7,0]       | [6,4] | [21,25] | 6000 | 1.941710<br>55 | 0.896546<br>93 | 172.9396<br>53 | 90.55572<br>897 |
| [40,7,8,3<br>0] | [2,9,3,0]       | [7,8] | [40,30] | 6000 | 1.948281<br>66 | 0.901766<br>5  | 93.71730<br>13 | 94.38333<br>919 |
| [29,6,2,4<br>9] | [8,9,0,0]       | [6,2] | [29,49] | 6000 | 1.918577<br>67 | 0.890594<br>34 | 196.4527<br>32 | 91.31910<br>003 |
| [26,3,24]       | [14,1,0]        | 3     | [26,24] | 6000 | 2.018965<br>62 | 0.939765<br>26 | 137.8080<br>17 | 96.34880<br>258 |
| [28,4,3,1<br>4] | [8,0,17,0<br>]  | [4,3] | [28,14] | 6000 | 1.987428<br>16 | 0.927547<br>97 | 180.6218<br>41 | 91.61488<br>031 |
| [3,9,8,15<br>]  | [0,0,5,0]       | [9,8] | [3,15]  | 6000 | 1.995718<br>65 | 0.935395<br>67 | 154.5263<br>79 | 93.08321<br>646 |
| [3,9,8,35<br>]  | [2,6,0,10<br>]  | [9,8] | [3,35]  | 6000 | 1.976607<br>68 | 0.927719<br>82 | 165.1249<br>42 | 95.93631<br>329 |
| [15,5,50]       | [8,6,0]         | 5     | [15,50] | 6000 | 2.026484<br>16 | 0.953830<br>98 | 131.6766<br>99 | 92.55934<br>971 |
| [37,2,6,2<br>7] | [0,6,0,4]       | [2,6] | [37,27] | 6000 | 1.926721<br>22 | 0.909460<br>83 | 125.1747<br>89 | 91.88999<br>526 |
| [48,9,1,2<br>4] | [1,0,0,0]       | [9,1] | [48,24] | 6000 | 1.997830<br>81 | 0.944911<br>08 | 139.4571<br>7  | 91.23838<br>306 |
| [24,6,1,4<br>9] | [9,0,5,0]       | [6,1] | [24,49] | 6000 | 2.013033<br>35 | 0.952930<br>76 | 108.2101<br>45 | 91.04820<br>068 |
| [48,4,6,3<br>1] | [7,0,0,0]       | [4,6] | [48,31] | 6000 | 2.021758<br>99 | 0.959777<br>29 | 80.43716<br>32 | 93.63920<br>876 |
| [41,9,7,4<br>7] | [0,0,0,0]       | [9,7] | [41,47] | 6000 | 2.054088<br>07 | 0.976697<br>67 | 24.20472<br>85 | 97.11339<br>655 |
| [30,4,35]       | [9,1,8]         | 4     | [30,35] | 6000 | 1.998072<br>22 | 0.950599<br>65 | 108.5957<br>54 | 91.13577<br>381 |
| [9,7,6,21<br>]  | [1,0,2,5]       | [7,6] | [9,21]  | 6000 | 1.926481<br>09 | 0.919580<br>22 | 133.8688<br>08 | 93.12559<br>345 |
| [36,7,36]       | [15,0,0]        | 7     | [36,36] | 6000 | 2.039111<br>94 | 0.974856<br>17 | 66.68654<br>95 | 91.33856<br>162 |
| [35,8,9,4<br>0] | [2,8,0,0]       | [8,9] | [35,40] | 6000 | 1.982424<br>63 | 0.950212<br>13 | 215.5042<br>32 | 94.11606<br>518 |
| [7,4,22]        | [0,16,0]        | 4     | [7,22]  | 6000 | 1.948478<br>48 | 0.935934<br>76 | 126.8866<br>15 | 92.25017<br>9   |
| [48,5,42]       | [6,13,0]        | 5     | [48,42] | 6000 | 1.950017<br>28 | 0.939045<br>48 | 74.52892<br>82 | 96.17340<br>004 |
| [40,5,3,4<br>8] | [0,10,0,0<br>]  | [5,3] | [40,48] | 6000 | 1.920429<br>74 | 0.927632<br>15 | 163.2558<br>36 | 95.66758<br>128 |
| [44,8,7,1<br>4] | [0,0,0,10<br>]  | [8,7] | [44,14] | 6000 | 1.911103<br>77 | 0.925241<br>86 | 211.3865<br>44 | 94.20383<br>91  |
| [5,6,8,19<br>]  | [10,2,13,<br>0] | [6,8] | [5,19]  | 6000 | 1.907584<br>51 | 0.925340<br>11 | 156.4996<br>7  | 93.12055<br>249 |



|                  |                 |        |         |      |                |                |                |                 |
|------------------|-----------------|--------|---------|------|----------------|----------------|----------------|-----------------|
| [1,3,4,16<br>]   | [22,0,4,0<br>]  | [3,4]  | [1,16]  | 6000 | 1.849491<br>3  | 0.900545<br>92 | 164.7769<br>95 | 93.25642<br>846 |
| [45,10,3,<br>17] | [0,0,10,1<br>0] | [10,3] | [45,17] | 6000 | 1.886698<br>34 | 0.922535<br>19 | 177.9038<br>57 | 91.05228<br>613 |
| [26,8,5,3<br>3]  | [2,2,0,0]       | [8,5]  | [26,33] | 6000 | 1.935342<br>19 | 0.949291<br>34 | 158.2307<br>14 | 93.23635<br>542 |
| [42,8,6,1<br>6]  | [12,11,0,<br>0] | [8,6]  | [42,16] | 6000 | 1.957072<br>29 | 0.961103<br>14 | 133.4594<br>31 | 96.93288<br>847 |
| [31,7,8,2<br>3]  | [8,0,0,0]       | [7,8]  | [31,23] | 6000 | 1.934967<br>87 | 0.953884<br>82 | 124.4485<br>6  | 94.37310<br>793 |
| [29,4,1,3<br>3]  | [6,0,0,4]       | [4,1]  | [29,33] | 6000 | 1.886727<br>11 | 0.931663<br>41 | 139.2977<br>81 | 95.49677<br>877 |
| [36,9,36]        | [2,0,0]         | 9      | [36,36] | 6000 | 1.928408<br>01 | 0.953903<br>4  | 147.6373<br>12 | 94.52075<br>289 |
| [20,10,2,<br>24] | [4,4,0,0]       | [10,2] | [20,24] | 6000 | 1.901141<br>6  | 0.941622<br>43 | 138.9170<br>18 | 93.78511<br>52  |
| [49,8,9,4<br>1]  | [9,8,0,7]       | [8,9]  | [49,41] | 6000 | 1.884624<br>12 | 0.934999<br>59 | 169.6958<br>28 | 94.94647<br>824 |
| [46,9,8,4<br>5]  | [14,0,9,0<br>]  | [9,8]  | [46,45] | 6000 | 1.816960<br>85 | 0.904649<br>25 | 145.7680<br>37 | 94.98594<br>83  |
| [27,6,3,4<br>7]  | [0,10,0,0<br>]  | [6,3]  | [27,47] | 6000 | 1.866643<br>63 | 0.932887<br>15 | 67.09650<br>45 | 87.65727<br>456 |
| [23,1,13]        | [16,8,8]        | 1      | [23,13] | 6000 | 1.907707<br>91 | 0.955051<br>29 | 150.4432<br>58 | 89.50570<br>707 |
| [8,1,9,20<br>]   | [5,0,6,0]       | [1,9]  | [8,20]  | 6000 | 1.790743<br>88 | 0.900514<br>74 | 158.0094<br>59 | 90.76763<br>209 |
| [43,3,17]        | [8,0,6]         | 3      | [43,17] | 6000 | 1.855006<br>17 | 0.934839<br>86 | 74.81645<br>14 | 91.02925<br>974 |
| [48,7,3,4<br>7]  | [0,0,9,0]       | [7,3]  | [48,47] | 6000 | 1.850588<br>67 | 0.934798<br>28 | 203.9194<br>31 | 92.52964<br>173 |
| [36,4,10,<br>28] | [3,9,0,0]       | [4,10] | [36,28] | 6000 | 1.831893<br>84 | 0.926977<br>36 | 221.8188<br>67 | 94.97831<br>421 |
| [21,1,6,2<br>2]  | [0,10,0,5<br>]  | [1,6]  | [21,22] | 6000 | 1.748503<br>28 | 0.886924<br>29 | 190.5864<br>04 | 92.85932<br>642 |
| [13,5,4,1<br>8]  | [17,0,0,0<br>]  | [5,4]  | [13,18] | 6000 | 1.806207<br>98 | 0.918433<br>26 | 153.6618<br>78 | 91.97022<br>388 |
| [20,1,6,3<br>7]  | [6,0,0,0]       | [1,6]  | [20,37] | 6000 | 1.699457<br>35 | 0.866458<br>26 | 215.4197<br>28 | 96.11993<br>111 |
| [2,6,27]         | [5,11,4]        | 6      | [2,27]  | 6000 | 1.869881<br>62 | 0.954817<br>01 | 104.1863<br>26 | 94.55015<br>01  |
| [42,6,10,<br>23] | [0,0,8,0]       | [6,10] | [42,23] | 6000 | 1.834576<br>3  | 0.939606<br>82 | 173.6339<br>23 | 96.67950<br>738 |
| [42,6,20]        | [11,7,9]        | 6      | [42,20] | 6000 | 1.818497<br>31 | 0.932727<br>99 | 142.6635<br>92 | 95.40364<br>908 |
| [39,7,5]         | [4,0,8]         | 7      | [39,5]  | 6000 | 1.847614<br>39 | 0.951410<br>07 | 110.5567<br>49 | 94.63590<br>198 |
| [17,7,5,4<br>1]  | [0,0,23,0<br>]  | [7,5]  | [17,41] | 6000 | 1.770365<br>83 | 0.913512<br>41 | 130.7034<br>13 | 94.23810<br>172 |
| [25,5,7,9<br>]   | [5,16,1,4<br>]  | [5,7]  | [25,9]  | 6000 | 1.815689<br>45 | 0.940011<br>14 | 210.5169<br>27 | 94.85669<br>237 |
| [17,10,2,<br>44] | [4,0,6,0]       | [10,2] | [17,44] | 6000 | 1.821028<br>76 | 0.946514<br>99 | 166.3954<br>76 | 96.42157<br>57  |
| [10,2,44]        | [7,10,0]        | 2      | [10,44] | 6000 | 1.824077<br>23 | 0.949670<br>16 | 136.2871<br>63 | 95.38888<br>438 |

|              |               |        |         |      |            |            |            |             |
|--------------|---------------|--------|---------|------|------------|------------|------------|-------------|
| [37,2,10,19] | [0,3,9,0]     | [2,10] | [37,19] | 6000 | 1.77877487 | 0.92850019 | 155.656852 | 94.10515487 |
| [4,2,49]     | [17,8,9]      | 2      | [4,49]  | 6000 | 1.79922599 | 0.9412311  | 143.421345 | 90.72886328 |
| [42,2,10,7]  | [0,7,7,0]     | [2,10] | [42,7]  | 6000 | 1.81973601 | 0.95625943 | 172.832513 | 92.28920712 |
| [18,2,10,9]  | [3,5,9,0]     | [2,10] | [18,9]  | 6000 | 1.78215716 | 0.93861654 | 197.684068 | 95.37510951 |
| [4,2,3,43]   | [11,0,6,0]    | [2,3]  | [4,43]  | 6000 | 1.73474624 | 0.91628757 | 190.2819   | 92.51667305 |
| [42,5,2,39]  | [14,0,0,5]    | [5,2]  | [42,39] | 6000 | 1.77492978 | 0.93992277 | 183.689786 | 94.23810172 |
| [43,5,38]    | [0,0,0]       | 5      | [43,38] | 6000 | 1.85782374 | 0.98640739 | 11.0819027 | 94.85669237 |
| [32,7,9,15]  | [0,0,9,3]     | [7,9]  | [32,15] | 6000 | 1.74022108 | 0.9248657  | 147.666528 | 96.4215757  |
| [38,7,1,33]  | [11,8,0,0]    | [7,1]  | [38,33] | 6000 | 1.75993268 | 0.93721314 | 148.660746 | 95.38888438 |
| [43,1,12]    | [24,11,0]     | 1      | [43,12] | 6000 | 1.73255382 | 0.92505196 | 148.375709 | 94.10515487 |
| [46,1,8,13]  | [14,10,1,0,0] | [1,8]  | [46,13] | 6000 | 1.72141946 | 0.92358123 | 111.118067 | 90.72886328 |
| [2,8,45]     | [0,4,0]       | 8      | [2,45]  | 6000 | 1.74366705 | 0.93949454 | 97.6128901 | 92.28920712 |
| [25,9,1,1]   | [2,14,0,0]    | [9,1]  | [25,1]  | 6000 | 1.73571191 | 0.93623677 | 190.628958 | 95.37510951 |
| [11,1,6,42]  | [0,13,9,0]    | [1,6]  | [11,42] | 6000 | 1.68875242 | 0.9133995  | 199.697913 | 92.51667305 |
| [26,1,48]    | [5,6,2]       | 1      | [26,48] | 6000 | 1.72692706 | 0.93676031 | 74.88457   | 96.4215757  |
| [10,3,4,45]  | [2,13,0,0]    | [3,4]  | [10,45] | 6000 | 1.70266918 | 0.92551434 | 149.616523 | 95.38888438 |
| [31,1,7,8]   | [0,2,3,0]     | [1,7]  | [31,8]  | 6000 | 1.72331083 | 0.93916832 | 153.033334 | 94.10515487 |
| [38,9,1,4]   | [0,0,0,0]     | [9,1]  | [38,4]  | 6000 | 1.80789938 | 0.98644168 | 17.309214  | 90.72886328 |
| [18,5,3,8]   | [6,4,0,0]     | [5,3]  | [18,8]  | 6000 | 1.67123879 | 0.91246663 | 188.952853 | 92.28920712 |

The simulation results shown in table 4.3 contains a 9-column architecture that contains the exact route, the number of hits each vehicle has faced from attackers, the identified RSUs, other vehicles in the list, attained throughput, PDR, Jitter & TDR produced in the certain simulation. It is quite clear from the table that one single vehicle is acting in a number of routes and multi-route management is implemented.

The proposed work also understands that a network overload may also produce a significant amount of delay that may lead to an understanding that the network or the vehicle has been hit and hence the proposed work does not consider every hit as a

network hit. To signify the secure and non-secure routes, the proposed work calculates the average of hit values. If the current hit is more than that of the average hit of the entire network, the route is said to be breached. For example, the average hit in the provided data in table 4.3 is 13.8 hits whereas the first route gets  $7+5+0+8=20$  hits. This route is considered to be breached at the first instance. In addition to this, the response time plays a vital role for the CHs when the broadcast has been made.

Table 4.4 Response Time

| 'Route'     | 'Hits'      | Average Response Time | Response time difference from the mean |
|-------------|-------------|-----------------------|--|
| [26,1,6,36] | [9,10,0,0]  | 2.68561756            | -2.8476824                             |
| [18,8,4,25] | [1,2,0,0]   | 9.49217272            | 3.95887272                             |
| [9,8,4,1]   | [4,0,0,0]   | 9.2741624             | 3.7408624                              |
| [26,5,4,7]  | [0,13,8,2]  | 2.3257386             | -3.2075614                             |
| [23,7,31]   | [14,9,0]    | 0.93708868            | -4.5962113                             |
| [21,6,4,25] | [6,0,7,0]   | 7.83676176            | 2.30346176                             |
| [40,7,8,30] | [2,9,3,0]   | 3.56426936            | -1.9690306                             |
| [29,6,2,49] | [8,9,0,0]   | 8.2915563             | 2.7582563                              |
| [26,3,24]   | [14,1,0]    | 5.32349613            | -0.2098039                             |
| [28,4,3,14] | [8,0,17,0]  | 9.89704233            | 4.36374233                             |
| [3,9,8,15]  | [0,0,5,0]   | 9.10125438            | 3.56795438                             |
| [3,9,8,35]  | [2,6,0,10]  | 6.70016765            | 1.16686765                             |
| [15,5,50]   | [8,6,0]     | 3.04103634            | -2.4922637                             |
| [37,2,6,27] | [0,6,0,4]   | 6.83237468            | 1.29907468                             |
| [48,9,1,24] | [1,0,0,0]   | 2.53909261            | -2.9942074                             |
| [24,6,1,49] | [9,0,5,0]   | 9.67172126            | 4.13842126                             |
| [48,4,6,31] | [7,0,0,0]   | 3.74079256            | -1.7925074                             |
| [41,9,7,47] | [0,0,0,0]   | 2.39515411            | -3.1381459                             |
| [30,4,35]   | [9,1,8]     | 4.81204359            | -0.7212564                             |
| [9,7,6,21]  | [1,0,2,5]   | 7.61880183            | 2.08550183                             |
| [36,7,36]   | [15,0,0]    | 1.11228031            | -4.4210197                             |
| [35,8,9,40] | [2,8,0,0]   | 9.11245167            | 3.57915167                             |
| [7,4,22]    | [0,16,0]    | 3.62410868            | -1.9091913                             |
| [48,5,42]   | [6,13,0]    | 5.26148942            | -0.2718106                             |
| [40,5,3,48] | [0,10,0,0]  | 1.03380004            | -4.4995                                |
| [44,8,7,14] | [0,0,0,10]  | 7.12099036            | 1.58769036                             |
| [5,6,8,19]  | [10,2,13,0] | 6.41970359            | 0.88640359                             |
| [1,3,4,16]  | [22,0,4,0]  | 4.60521815            | -0.9280819                             |

|              |             |            |            |
|--------------|-------------|------------|------------|
| [45,10,3,17] | [0,0,10,10] | 6.205295   | 0.671995   |
| [26,8,5,33]  | [2,2,0,0]   | 5.91588534 | 0.38258534 |
| [42,8,6,16]  | [12,11,0,0] | 3.36092179 | -2.1723782 |
| [31,7,8,23]  | [8,0,0,0]   | 2.67217581 | -2.8611242 |
| [29,4,1,33]  | [6,0,0,4]   | 3.20821006 | -2.3250899 |
| [36,9,36]    | [2,0,0]     | 0.86513283 | -4.6681672 |
| [20,10,2,24] | [4,4,0,0]   | 9.93658401 | 4.40328401 |
| [49,8,9,41]  | [9,8,0,7]   | 0.40229002 | -5.13101   |
| [46,9,8,45]  | [14,0,9,0]  | 6.0076522  | 0.4743522  |
| [27,6,3,47]  | [0,10,0,0]  | 8.95243522 | 3.41913522 |
| [23,1,13]    | [16,8,8]    | 6.65185406 | 1.11855406 |
| [8,1,9,20]   | [5,0,6,0]   | 8.6719847  | 3.1386847  |
| [43,3,17]    | [8,0,6]     | 8.37525444 | 2.84195444 |
| [48,7,3,47]  | [0,0,9,0]   | 7.06454655 | 1.53124655 |
| [36,4,10,28] | [3,9,0,0]   | 8.23218389 | 2.69888389 |
| [21,1,6,22]  | [0,10,0,5]  | 1.60520519 | -3.9280948 |
| [13,5,4,18]  | [17,0,0,0]  | 0.45539484 | -5.0779052 |
| [20,1,6,37]  | [6,0,0,0]   | 9.37327889 | 3.83997889 |
| [2,6,27]     | [5,11,4]    | 1.58481161 | -3.9484884 |
| [42,6,10,23] | [0,0,8,0]   | 9.22669208 | 3.69339208 |
| [42,6,20]    | [11,7,9]    | 5.05634602 | -0.476954  |
| [39,7,5]     | [4,0,8]     | 6.38185498 | 0.84855498 |
| [17,7,5,41]  | [0,0,23,0]  | 8.12775908 | 2.59445908 |
| [25,5,7,9]   | [5,16,1,4]  | 3.06130171 | -2.4719983 |
| [17,10,2,44] | [4,0,6,0]   | 8.87463055 | 3.34133055 |
| [10,2,44]    | [7,10,0]    | 3.60705109 | -1.9262489 |
| [37,2,10,19] | [0,3,9,0]   | 2.81245206 | -2.7208479 |
| [4,2,49]     | [17,8,9]    | 9.90237475 | 4.36907475 |
| [42,2,10,7]  | [0,7,7,0]   | 4.84891426 | -0.6843857 |
| [18,2,10,9]  | [3,5,9,0]   | 7.39639062 | 1.86309062 |
| [4,2,3,43]   | [11,0,6,0]  | 3.26090588 | -2.2723941 |

The response time is the time when a CH replies to the broadcast request of any other CH. For example, the average response time for the mentioned route is 5.5 milliseconds and a negative deviation shows that the route is delivering data faster than the average response time. A positive response time difference shows that the route might be suffering any network distortion. To illustrate further, the data has been segmented into different clusters and is defined in the next chapter. These simulation parameters are further used for the analysis of the evaluation of the trust worthy nodes and the nodes that

are not reliable in the network. In order to evaluate, statistical machine learning is used which is discussed in the proceeding chapters.

### 4.3 Summary of the Chapter

This chapter introduces the V2V communication parameters. An AODV-based architecture has been created using MATLAB. The deployment node, communication architecture, and simulation properties have been described. The pseudo-code for the behavior of nodes is also explained well in the presented chapter. The simulated data is presented and a mechanism is discussed to differentiate between trusted and untrusted nodes.

## Chapter 5: Labelling and Classification

---

The chapter discusses the process of labeling and classification to identify how reliable the nodes are so that a trust worthy V2V communication can be presented. The integration of fuzzy rule set and neural network architecture is presented while discussing its design, and implementation, followed by the performance analysis.

### 5.1 Introduction to the Architecture

The packet efficiency of any routing architecture depends upon how reliable the nodes are that are participating in the route discovery mechanism. As the proposed solution uses a broadcast-oriented mechanism, the chances of the route getting intruded are more probabilistic than the route that follows certain path architecture. Due to the limitations of the fixed route architectures, jamming of a network, increase in latency, and the issue of overload, the only solution for this modern world is dynamic routing. Now it becomes essential to justify the route based on the behaviour and architecture of the node which it exhibits during the data transfer. The efficacy of the route can be evaluated utilizing Quality of Service (QoS) parameters [108]. The QoS parameters are completely based on the evaluation of the current scenario where the evaluation is taking place. It has been already significantly justified in the earlier chapters that the proceeding for further development will be based on the QoS parameters. As the QoS parameters are different and represent different ranges of values, it is required to develop mathematical expression architecture. Manual interpretations for such long expressions and values are almost impossible for the desired set of time and hence it becomes compulsory to train a system for the calculations. This will not only reduce the human computation effort but it will also enhance the performance of the overall system due to its fast calculation mechanism. Training a system for any specific goal and then getting analysis for the computation of the goal is called machine learning [109].

The utilization of machine learning has always been an area of interest for researchers and industrialists due to its sophistication and the ways it can be tweaked to get

maximum benefit [110-114]. As the network is error-prone and the simulation design has been implemented in such a way that multiple security threats have to be identified, a separation algorithm is required viz. a clustering algorithm is to be implemented.

As the foremost requirement is to separate the supplied data, supervised clustering will be applied to the route dataset containing the following QoS parameters.

- a) Throughput: It refers to the total number of collected data-packets in a given interval of time. If arranged mathematically, the throughput can be illustrated using eq(5.1)

$$Throughput = \frac{\int_{t=0}^{t'} Rec_{packets} dt}{t} \quad (5.1)$$

Where t is the total amount of supplied time for the simulation

- b) Packet Delivery Ratio (PDR): It refers to the ratio of the total attained packets at the receiver end to the total number of supplied packets at the transmitter end. The mathematical expression is quite simple and is expressed by equation 5.2

$$PDR = \frac{\int_{t=0}^{t'} \frac{Attained\ at\ transmitter\ end}{Sent\ at\ transmitter\ end} dt}{t} \quad (5.2)$$

- c) Jitter: It is the total amount of propagated delay in the network for all the transmissions from the transmitter end to the receiver end. Mathematical expression for such architecture is also quite simple and is expressed by eq 5.3.

$$Jitter = \oint_{t=1}^{t'} Delay_{route} \quad (5.3)$$

- d) TDR: True detection rate is the % of successful packet transmissions made during the data transmission. It is computed using the following relation.

$$TDR = \int_{t=1}^{t'} Number\ of\ Successful_{packe\ transmission} \quad (5.4)$$

ML performs separation by incorporating these four parameters altogether. The dataset which is attained by the simulation contains the route information as well as drop

information. If a node is performing below the threshold, it is noted that the node has been hit by any intruder as physical faults are not considered in the proposed case scenario. ML performs the separation of the data as shown in Figure 5.1.

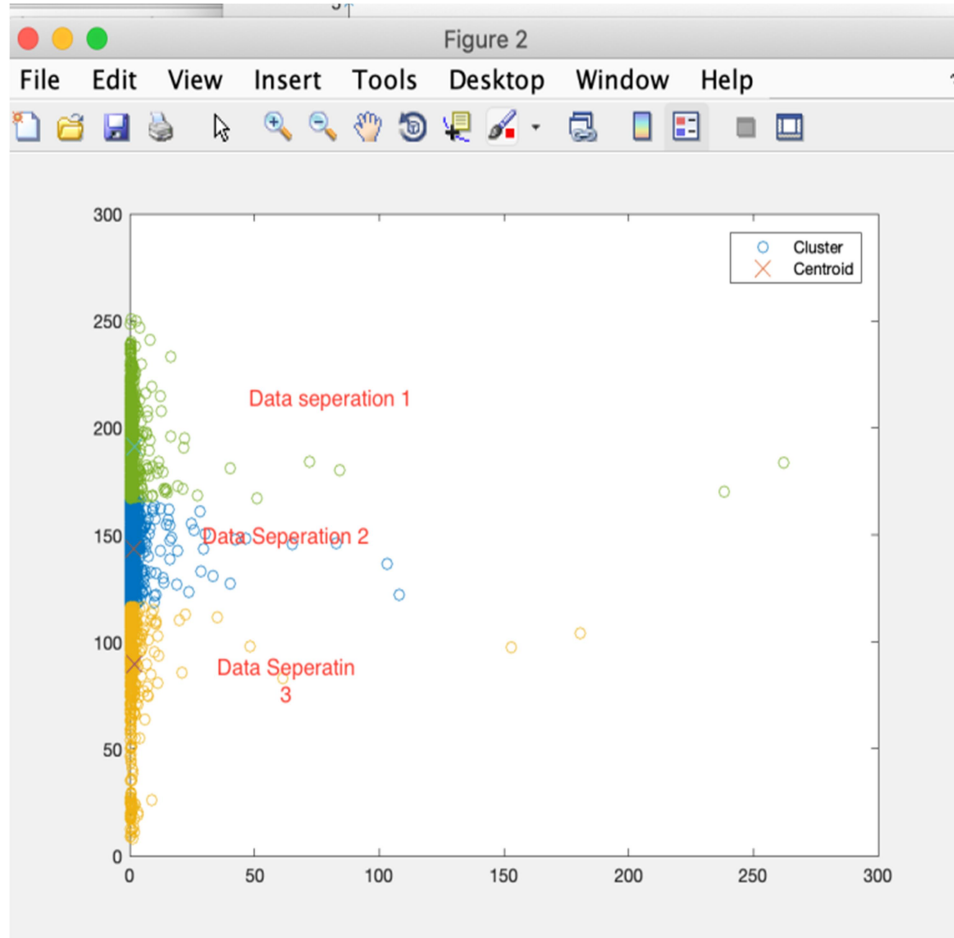


Figure 5.1 Data Separation Using Machine Learning

The proposed algorithm architecture considers multiple threats in the network and to label them, a statistical approach has been utilized. It has been significantly described in the literature survey that, a DDoS attack has the most impact when it comes to distorting the data pattern or harming the route discovery architecture. Other than DDoS, replay attack is also employed in the simulation architecture, and the rest of the attacks are categorized as the third clustered element. The proposed algorithm uses Mean Squared Error (MSE) for the calculation of cluster element's co-relation. The following rule set has been applied for the construction of the labels.



- I. Fetch the cluster of each route value
- II. Arrange routes as per their cluster index
- III. Evaluate MSE for each parameter utilized for the implementation of k-means
- IV. Take the moving average mean
- V. Repeat the process for every cluster
- VI. Find the maximum MSE out of the clusters.
- VII. Label Maximum MSE as a DDoS class
- VIII. Find second largest MSE
- IX. Label cluster as replay
- X. Find least MSE
- XI. Label the cluster as others.

Figure 5.2 shows the simulation architecture and demonstrates the work procedure of MATLAB for the same.

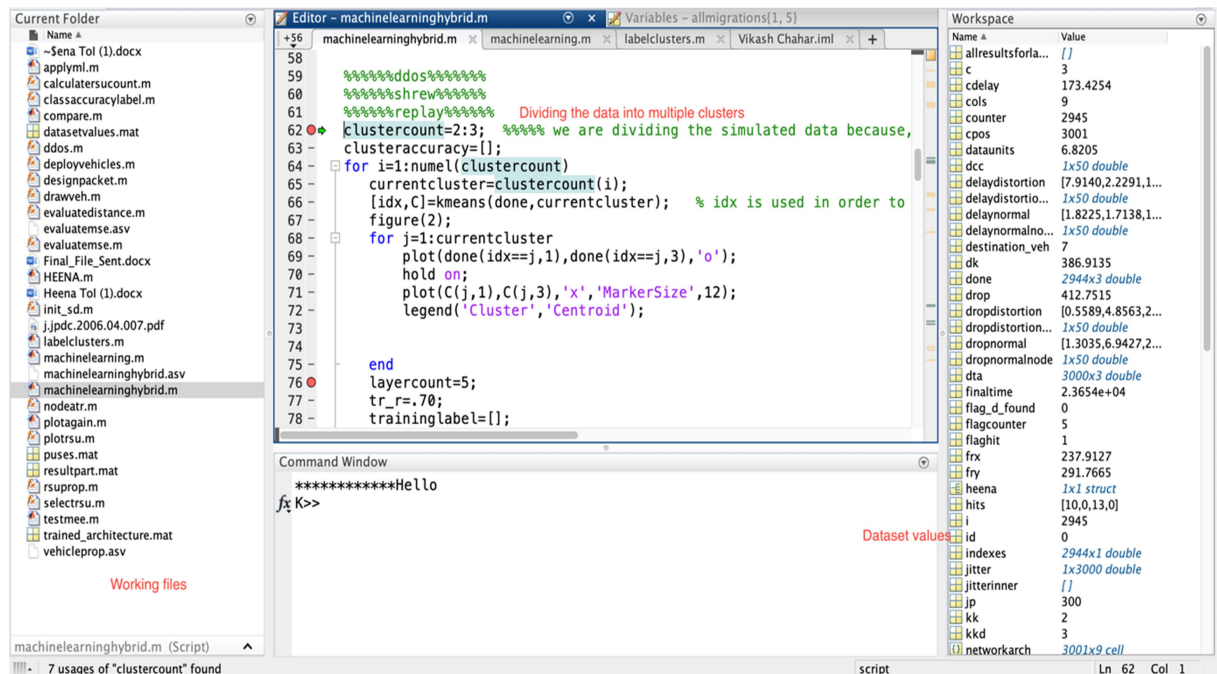


Figure 5.2 Implementation Design for Separation of the Data

The proposed simulation has been done for 10,000 consecutive rounds to collect the data and it is not possible to demonstrate all the data hence a sample record is presented in Table 5.1. It is the same set of data that has been generated using the dragonfly algorithm in the previous chapter. The aim is to partition the data based on the QoS parameters in good moderate and bad routes. The good routes will be termed as trusted roots and the nodes involved in the root discovery will be termed as trusted nodes. It is also possible that a node may fall into both the trusted and un-trusted group as one vehicle or node is involved in multiple communications. To separate the records into multiple groups, the proposed work has utilized the k-means algorithm as it is a widely used clustering algorithm that has several advantages over the fuzzy c-means and k-medoids algorithms. Here are some of the key advantages of the k-means algorithm:

- **Simplicity and Efficiency:** K-means is a relatively simple and easy-to-understand algorithm. It is computationally efficient and can handle large datasets with a large number of dimensions. The simplicity of the algorithm makes it easier to implement and interpret the results.
- **Speed:** K-means is generally faster than fuzzy c-means and k-medoids algorithms, especially for large datasets. This is because the k-means algorithm has a time complexity of  $O(nkdi)$ , where  $n$  is the number of data points,  $k$  is the number of clusters,  $d$  is the number of dimensions, and  $i$  is the number of iterations. In practice, k-means often converge in a small number of iterations.
- **Hard Cluster Assignments:** K-means produces hard cluster assignments, which means that each data point is assigned to a single cluster. This can be advantageous in scenarios where the data points are expected to belong to distinct and non-overlapping clusters. It provides clear boundaries between clusters, making it easier to interpret and analyse the results.
- **Cluster Centroids:** K-means represents each cluster by its centroid, which is the mean of all the data points assigned to that cluster. The centroid is a representative point within the cluster and can be used for further analysis. It provides a compact and interpretable representation of the cluster.

- **Deterministic Results:** K-means algorithm produces deterministic results, meaning that running the algorithm multiple times with the same input will always yield the same output (assuming the same initializations). This can be advantageous for the reproducibility and consistency of results.

Table 5.1 Affected Discoveries According to K-Means

| Jitter     | PDR        | Throughput | TDR         |
|------------|------------|------------|-------------|
| 262.204302 | 0.92034747 | 183.827065 | 92.86305972 |
| 238.259239 | 0.9463236  | 170.153141 | 95.48405124 |
| 180.492451 | 0.94584096 | 104.027562 | 95.43535286 |
| 152.861949 | 0.97567778 | 97.2571782 | 98.445888   |
| 107.843039 | 0.89747997 | 122.043607 | 90.55572897 |
| 103.063749 | 0.93541466 | 136.525848 | 94.38333919 |
| 84.1667947 | 0.90504559 | 180.158819 | 91.31910003 |
| 82.8792883 | 0.95489398 | 146.099295 | 96.34880258 |
| 72.1288947 | 0.90797701 | 183.981697 | 91.61488031 |
| 65.0960157 | 0.9225294  | 145.663094 | 93.08321646 |
| 61.2760114 | 0.95080588 | 82.9675297 | 95.93631329 |
| 50.8915449 | 0.91733746 | 167.267342 | 92.55934971 |
| 48.0610437 | 0.91070362 | 97.9572271 | 91.88999526 |
| 46.4261138 | 0.90424562 | 148.204267 | 91.23838306 |
| 42.1943168 | 0.90236076 | 147.683476 | 91.04820068 |
| 40.2546389 | 0.92803973 | 127.405037 | 93.63920876 |
| 40.1405603 | 0.96247172 | 180.977223 | 97.11339655 |
| 35.2234983 | 0.90322868 | 111.463125 | 91.13577381 |
| 33.2400746 | 0.92294939 | 130.88548  | 93.12559345 |
| 30.3214382 | 0.90523847 | 150.287926 | 91.33856162 |
| 29.474536  | 0.93276576 | 143.588911 | 94.11606518 |
| 28.1531964 | 0.91427333 | 161.134888 | 92.250179   |

|            |            |            |             |
|------------|------------|------------|-------------|
| 28.5904415 | 0.9531556  | 132.812434 | 96.17340004 |
| 27.1078524 | 0.94814253 | 168.191738 | 95.66758128 |
| 25.919839  | 0.93363567 | 152.356392 | 94.2038391  |
| 24.8189412 | 0.92289943 | 155.063149 | 93.12055249 |
| 23.502487  | 0.92424607 | 123.319209 | 93.25642846 |
| 21.6642161 | 0.90240125 | 190.619212 | 91.05228613 |
| 21.9363707 | 0.92404713 | 194.901687 | 93.23635542 |
| 22.1331896 | 0.96068274 | 112.785072 | 96.93288847 |
| 21.145125  | 0.93531326 | 171.560427 | 94.37310793 |
| 20.8439964 | 0.94644974 | 85.6678712 | 95.49677877 |
| 19.9876733 | 0.93677654 | 110.200874 | 94.52075289 |
| 19.2802389 | 0.92948578 | 172.75192  | 93.7851152  |
| 19.1888761 | 0.94099582 | 142.364721 | 94.94647824 |
| 18.7330393 | 0.941387   | 126.823751 | 94.9859483  |
| 16.8102091 | 0.86875396 | 148.487008 | 87.65727456 |
| 16.5358114 | 0.88707341 | 233.071639 | 89.50570707 |
| 16.5531397 | 0.8995801  | 195.984548 | 90.76763209 |
| 16.2107672 | 0.90217304 | 138.511279 | 91.02925974 |
| 16.0155836 | 0.91704303 | 147.532776 | 92.52964173 |
| 16.0730504 | 0.94131134 | 154.290501 | 94.97831421 |
| 15.495028  | 0.92031047 | 157.106459 | 92.85932642 |
| 15.0308505 | 0.91149875 | 169.545356 | 91.97022388 |
| 15.5814465 | 0.95262568 | 161.781101 | 96.11993111 |
| 14.7935156 | 0.93706789 | 171.134661 | 94.5501501  |
| 14.7856257 | 0.95817153 | 155.004923 | 96.67950738 |
| 14.4341454 | 0.94552675 | 171.655924 | 95.40364908 |
| 14.0629202 | 0.93791776 | 170.077613 | 94.63590198 |
| 13.746595  | 0.93397524 | 171.489116 | 94.23810172 |
| 13.6715078 | 0.94010597 | 127.615356 | 94.85669237 |

|            |            |            |             |
|------------|------------|------------|-------------|
| 13.4648395 | 0.95561522 | 129.74462  | 96.4215757  |
| 13.1997398 | 0.94538042 | 179.221868 | 95.38888438 |
| 12.6653271 | 0.93265763 | 207.769205 | 94.10515487 |
| 12.104089  | 0.89919587 | 214.993766 | 90.72886328 |

The entire table can be represented using Figure 5.3.

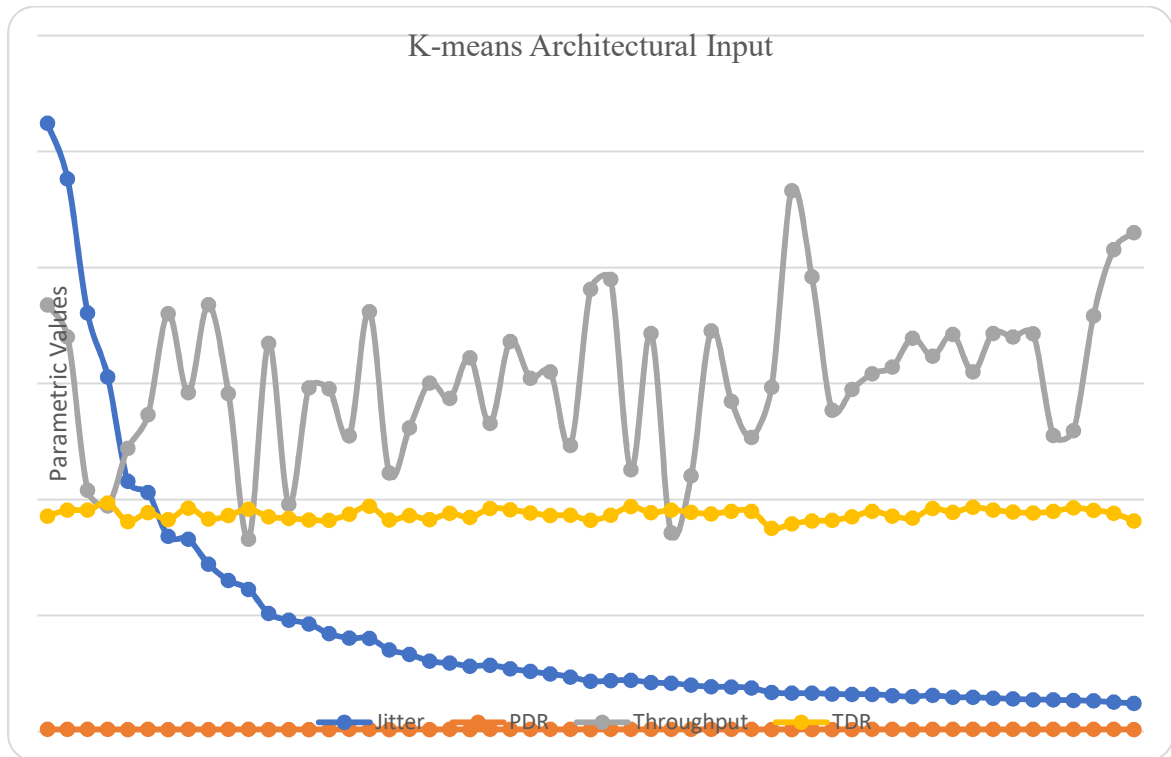


Figure 5.3 Architectural Input to the K-Means

The evaluated class architecture is passed to multiple classifiers in order to see the precision of the network segregation. Though any multi-class classifier could have been applied by looking at the computation complexity of the proposed work and data size, Neural Networks are one of the finest algorithm architectures that can be applied to train and classify the system. In order to validate the data, other classifiers have also been used and have been illustrated in the result section of this dissertation. In order to be precise on the proposed work architecture for labelling, the work flow is described in Figure 5.4.

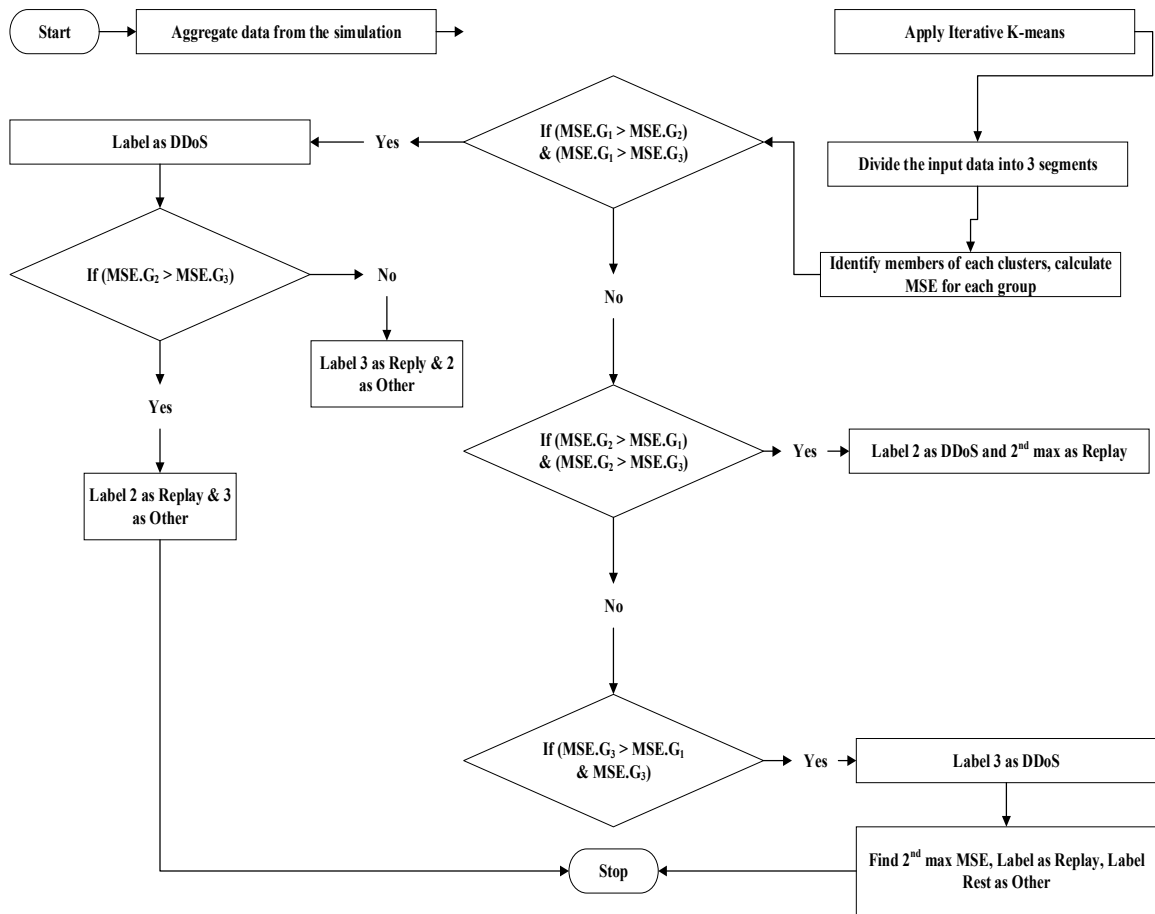


Figure 5.4 Labelling using MSE

The work architecture of the separation does illustrate the labelling or naming convention via MSE only. The labelling convention again requires an architectural explanation of statistical machine learning. To enhance the labelling pattern efficiency and get more refined results, the proposed algorithm architecture utilizes a fuzzy rule base mechanism system that considers Standard Error alongside Mean Squared Error. The entire simulation has been done over MATLAB and hence all the toolboxes that are supported under machine learning architecture are available for MATLAB 2016 onwards. Two parameter set of Standard Error (SE) and Mean Squared Error (MSE) has been used as follows.

## 5.2 Fuzzy Logic

A fuzzy logic (FL) system can manage numerical data and handle linguistic information at the same time. A FL is a nonlinear mapping of a scalar output (the vector output) from input data (feature). The importance of FL is that it is so diverse. There are a plethora of scenarios that can lead to the realization of different mappings. This richness necessitates a thorough understanding of FL and the components that make it up a Fuzzy Logic System (FLS). Anonymously, this is comparable to solving problems in engineering; engineers are always confronted with the challenge of representation. The nonlinear mapping's specifics are established by fuzzy set theory (FST) and FL. It accomplishes this by illustrating how crisp set theory and dual logic may be extended to their fuzzy counterparts. Additionally, FL is concerned with Modes of communication that are approximate. In general, logically implies that Fuzzy logic reasoning chains are brief in length, and rigor is less crucial than it is in traditional logical systems. In other words, fuzzy logic is a type of reasoning that is based on uncertainty. Fuzzy has a larger expressive power. The fact that it has logic is what gives it its name. FL is the formal foundation of approximate reasoning, with exact reasoning being considered a limiting instance.

The FLS is in which imprecise data and vague statements are fed as input and decisions on that statement are considered as output.

MATLAB is a programming language that is of high performance and used for the computation of technical problems. For FL, MATLAB is used for computation, processing, implementation, visualization, and coding in an easy environment in which problems have been expressed using the fuzzy logic sets.

### 5.2.1 Utilization of Fuzzy Logic in the Proposed Work Frame

The proposed algorithm utilizes a fuzzy logic toolbox for the implementation of the fuzzy-oriented architecture in the system. The proposed algorithm uses Mamdani architecture to create the rule base as shown in Figure 5.5.

### 5.2.2 Root Mean Square Error (RMSE)

It is the standard deviation of the predicted errors observed during effort estimation. It computes the distribution distance with respect to the regression line. To verify the experimental outcomes it is computed against the desired output and the estimated output of the project using the following equation.

$$RMSE = \sqrt{(E_{predicted} - E_{known})^2} \quad (5.5)$$

### 5.2.3 Standard Error (SE)

It is a statistical term that is used to measure accuracy with the help of sample distribution for representing a population by using standard deviation. It is used to refer SD of different sample statistics, like mean or median. To compute the value of standard error the following equation is used.

$$SE = \frac{\sigma}{\sqrt{n}} \quad (5.6)$$

Where SE defines as the standard error of the sample, n defines as the number of samples and  $\sigma$  defines as the sample standard deviation.

Both these parameters have been passed to the fuzzy inference rule engine in order to create a rule set for the labelling of the clusters. In order to operate through the fuzzy inference engine, the proposed algorithm architecture uses the Fuzzy logic toolbox of MATLAB. The fuzzy logic toolbox comes with two types of inference engines namely “Mamdani” and “ Sugeno“ as shown in Figure 5.5.[115]. The proposed work has opted Mamdani inference engine due to its lower computation complexity.

Two input variables namely the MSE and the SE have been passed and each input contains three membership functions namely Random Distribution, Moderate Distribution, and Close distribution. The illustration is shown in figure 5.8. There is one output variable viz the closeness. The proposed algorithm considers two major attacks namely DDoS and Replay, whereas one minor attack could be any server attack whose intensity is lower than DDoS and Replay. The output has also three member ship



functions namely low, moderate, and high. The range for every membership function has been kept between 0-1.

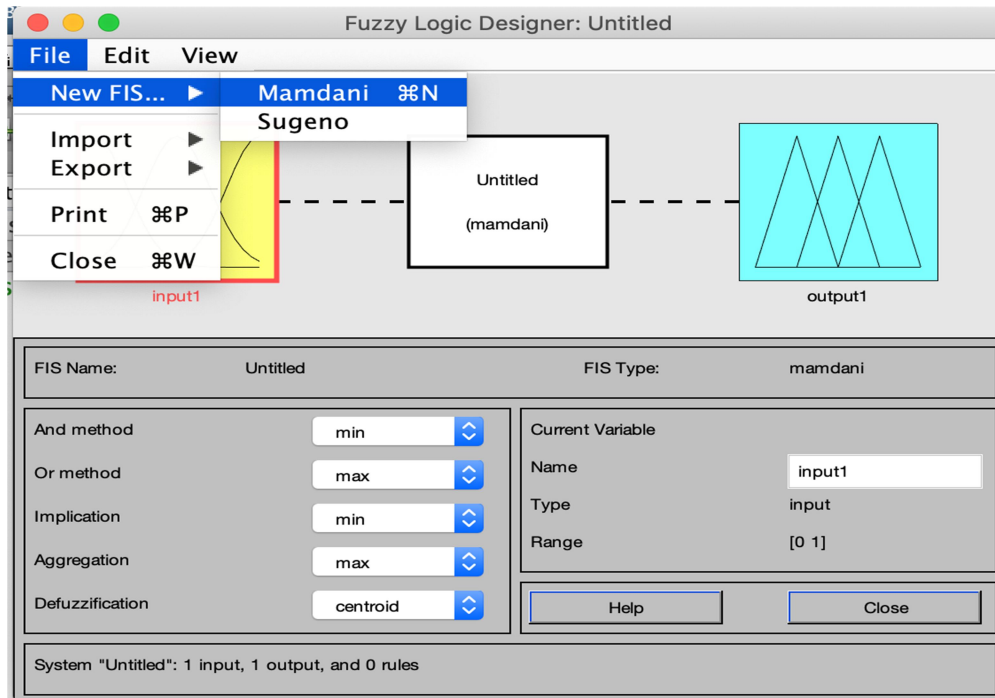


Figure 5.5 Fuzzy Inference Engine Models

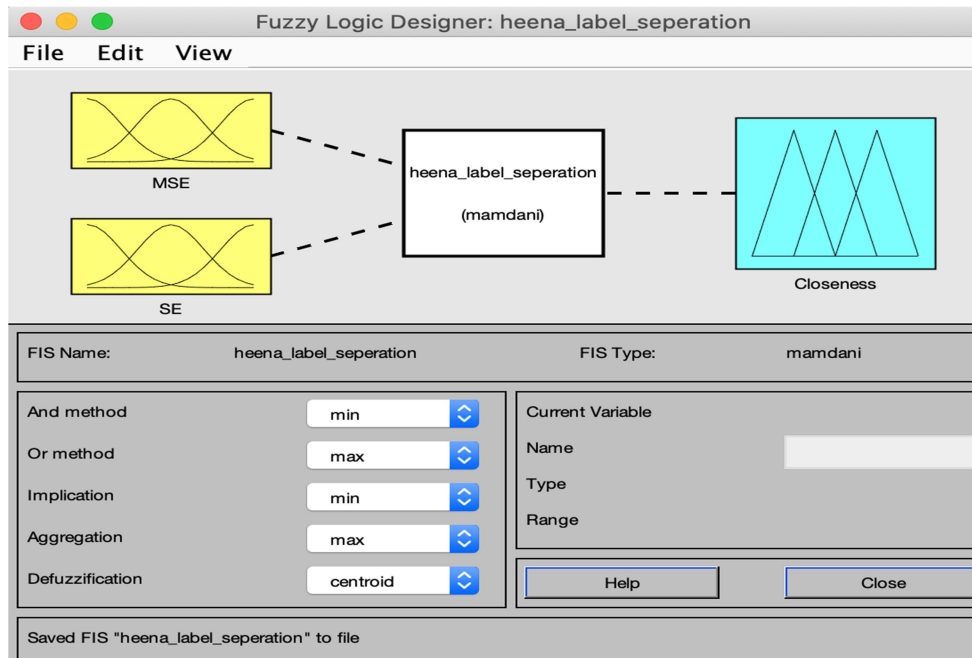


Figure 5.6 Fuzzy Logic Creation

Each input and output has three membership functions and a total of 6 rules have been created. The membership function of each input and output is illustrated in Figures 5.7 and 5.8 as follows.

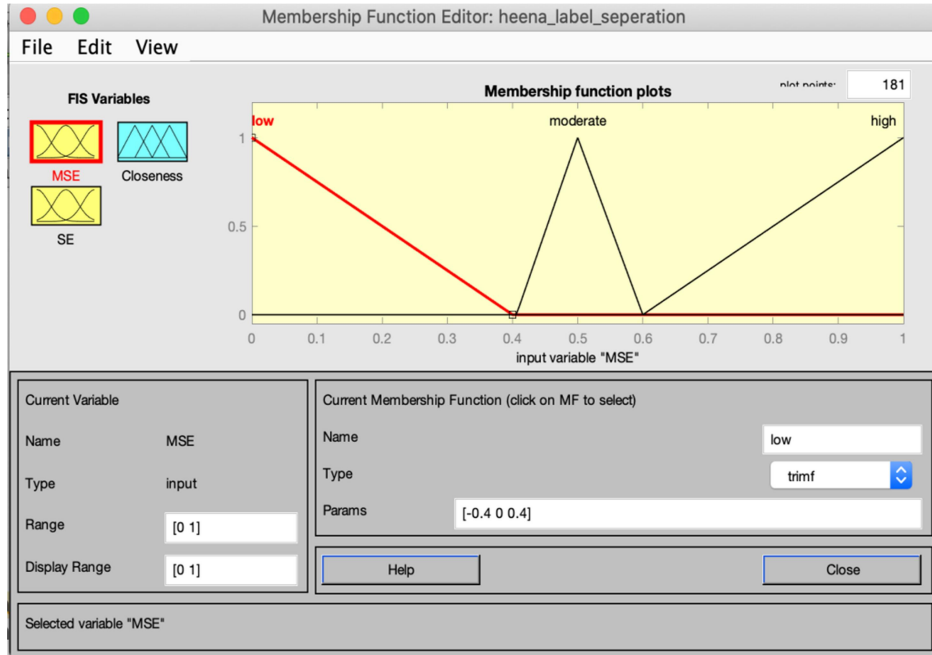


Figure 5.7 Input Membership Function

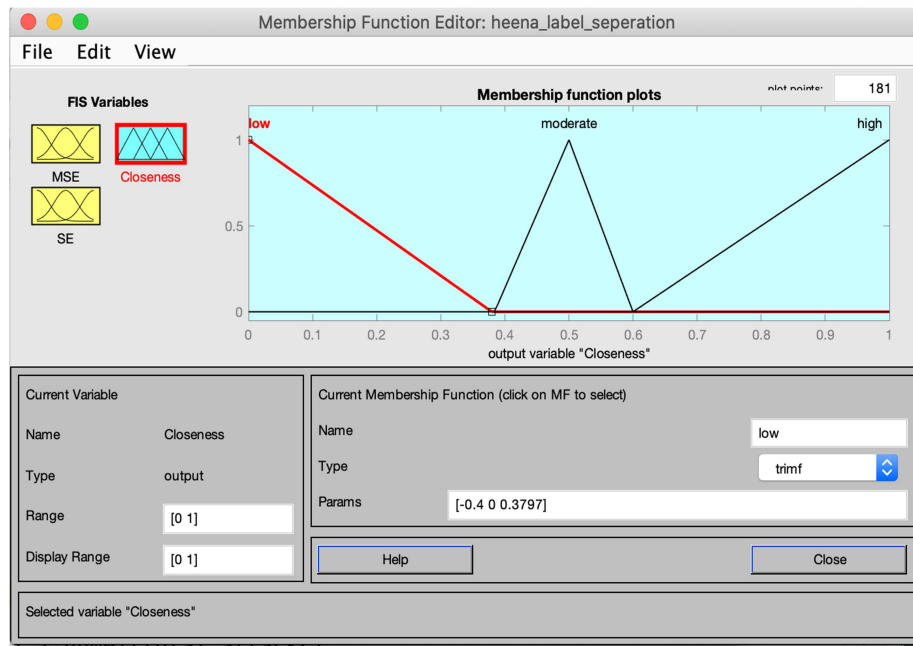


Figure 5.8 Output Membership Function

6 rules have been formatted utilizing the membership functions. The membership functions have been used to form additive rules. To illustrate the fuzzy architecture, the following pseudo-code is sufficient enough but still pictorial representations have been supplied in the later section of the report.

1. *[System]*
2. *Name = 'heena\_label\_seperation'*
3. *Type = 'mamdani'*
4. *Version = 2.0*
5. *NumInputs = 2*
6. *NumOutputs = 1*
7. *NumRules = 6*
8. *AndMethod = 'min'*
9. *OrMethod = 'max'*
10. *ImpMethod = 'min'*
11. *AggMethod = 'max'*
12. *DefuzzMethod = 'centroid'*
13. *[Input1]*
14. *Name = 'MSE'*
15. *Range = [0 1]*
16. *NumMFs = 3*
17. *MF1 = 'low': 'trimf', [-0.4 0 0.4]*
18. *MF2 = 'moderate': 'trimf', [0.406 0.5 0.599898063200816]*
19. *MF3 = 'high': 'trimf', [0.6 1 1.4]*
20. *[Input2]*
21. *Name = 'SE'*

22.  $Range = [0\ 1]$
23.  $NumMFs = 3$
24.  $MF1 = 'low': 'trimf', [-0.4\ 0\ 0.4]$
25.  $MF2 = 'moderate': 'trimf', [0.396\ 0.5\ 0.693679918450561]$
26.  $MF3 =$   
 $'high': 'trimf', [0.685626911314985\ 1.08562691131498\ 1.48562691131498]$
27.  $[Output1]$
28.  $Name = 'Closeness'$
29.  $Range = [0\ 1]$
30.  $NumMFs = 3$
31.  $MF1 = 'low': 'trimf', [-0.4\ 0\ 0.379714576962283]$
32.  $MF2 = 'moderate': 'trimf', [0.384\ 0.5\ 0.599898063200816]$
33.  $MF3 = 'high': 'trimf', [0.6\ 1\ 1.4]$
34.  $[Rules]$
35.  $1\ 1,1\ (1): 1$
36.  $1\ 2,2\ (1): 1$
37.  $2\ 2,2\ (1): 1$
38.  $2\ 3,3\ (1): 1$
39.  $3\ 3,3\ (1): 1$
40.  $3\ 2,3\ (1): 1$

Based on the formatted architecture and rulesets, the following rule viewer has depicted the nature of the system and is illustrated by Figures 5.9, 5.10, 5.11, 5.12, 5.13, and 5.14.

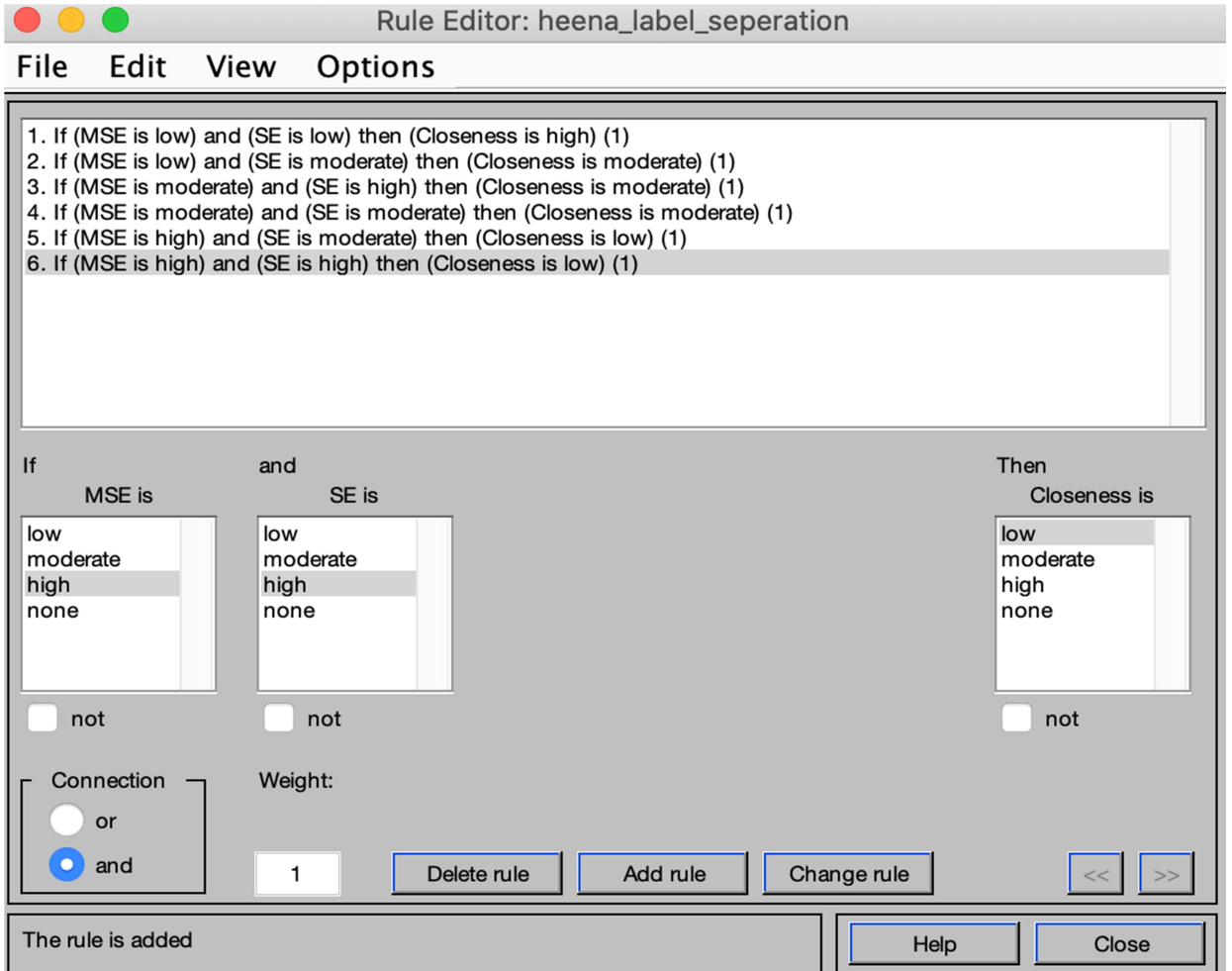


Figure 5.9 The Rule Viewer

As illustrated in the pseudo-code of the Fuzzy logic formation, 6 rules have been created and are presented in Figure 5.9. The rules are additive in nature and support AND operation throughout. The rule viewer shows different values of MSE and SE and their predictions regarding the closeness in the set.

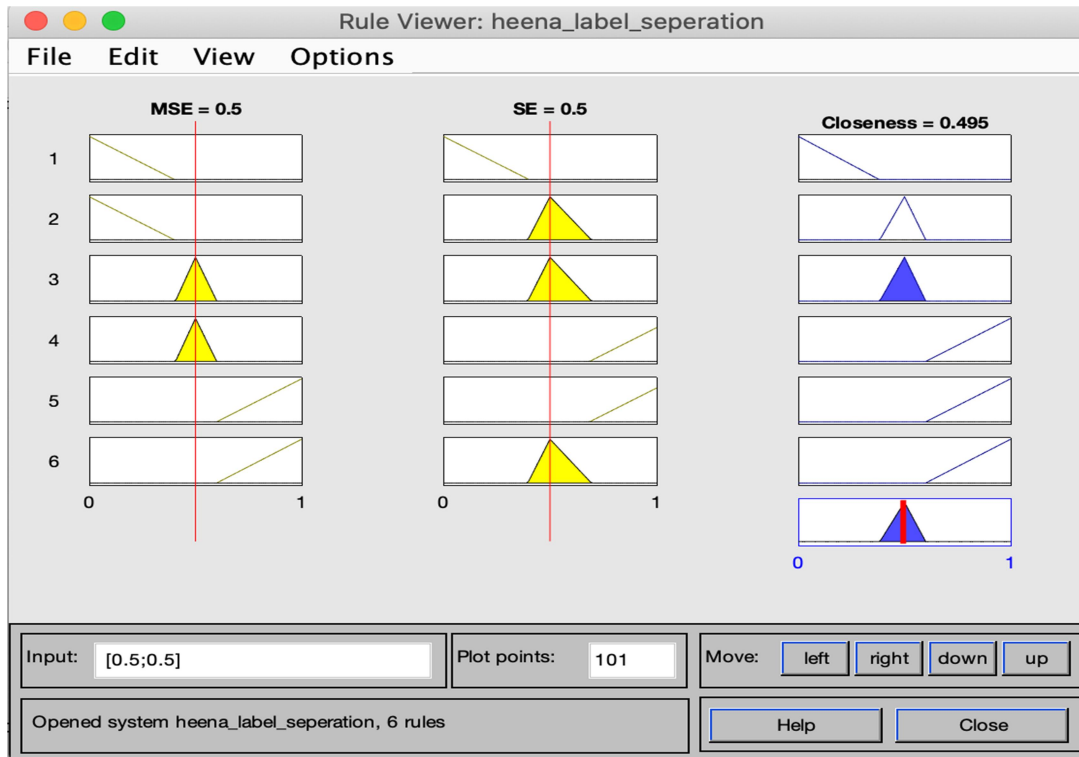


Figure 5.10 Moderate MSE and Moderate SE Resulting into Moderate Closeness

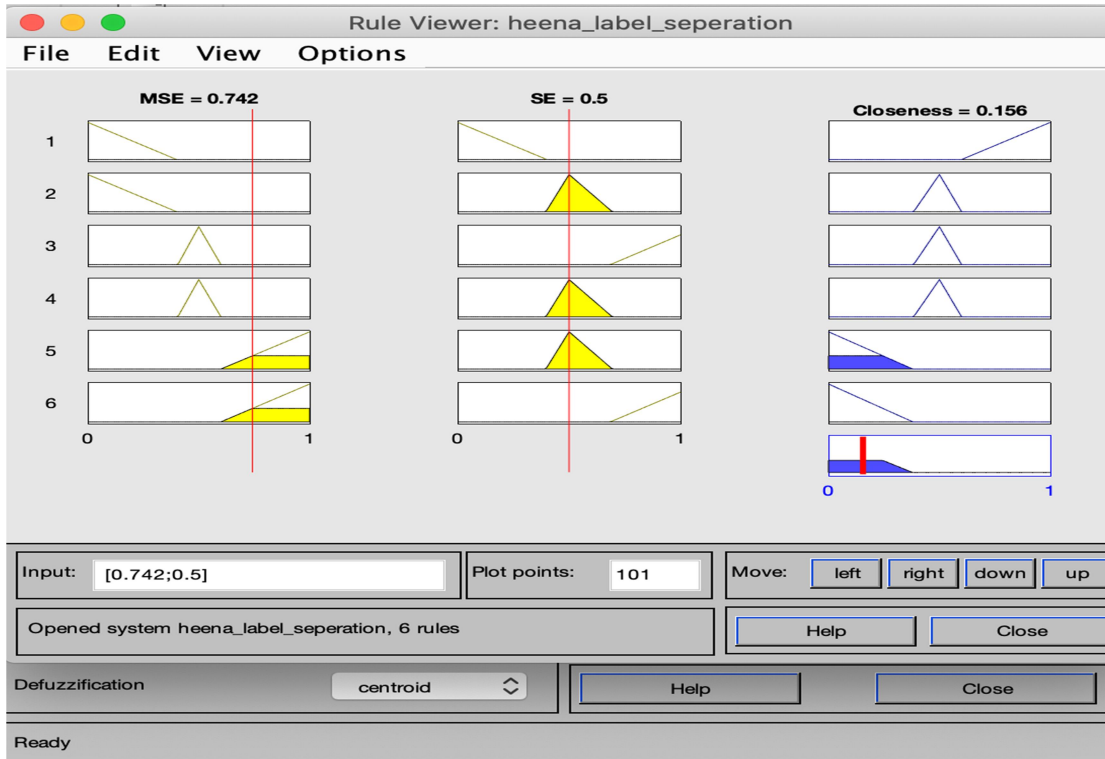


Figure 5.11 High MSE and Moderate SE Resulting into Low Closeness

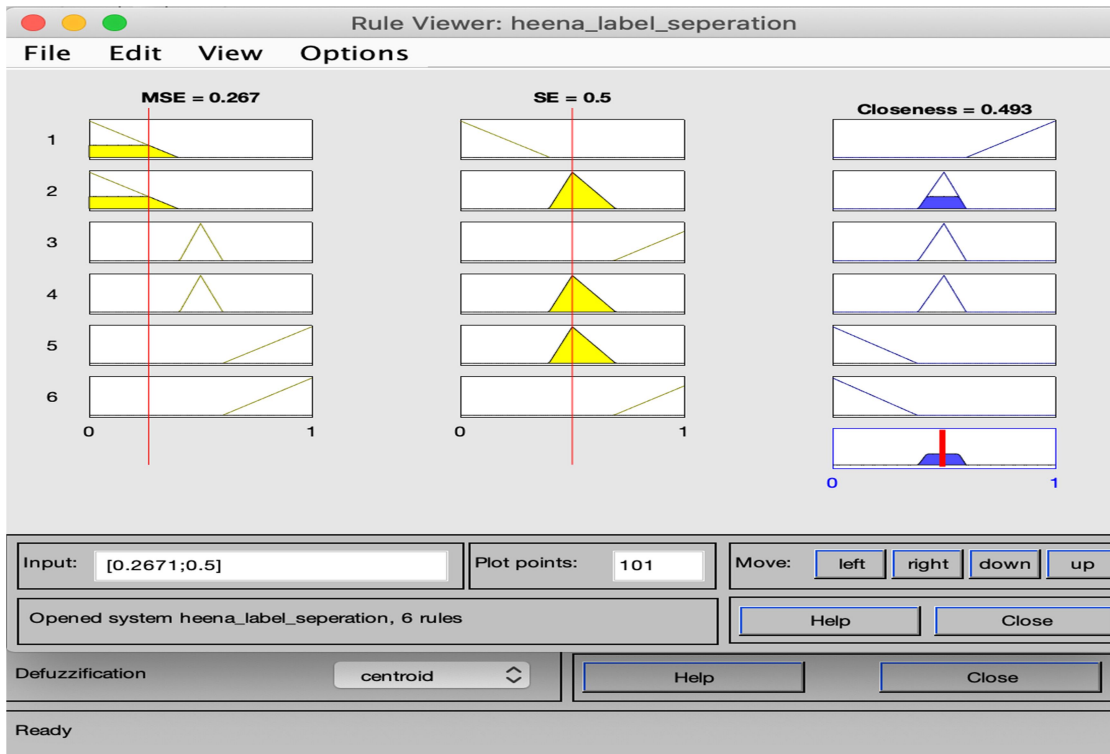


Figure 5.12 Low MSE and Moderate SE Resulting into Moderate Closeness

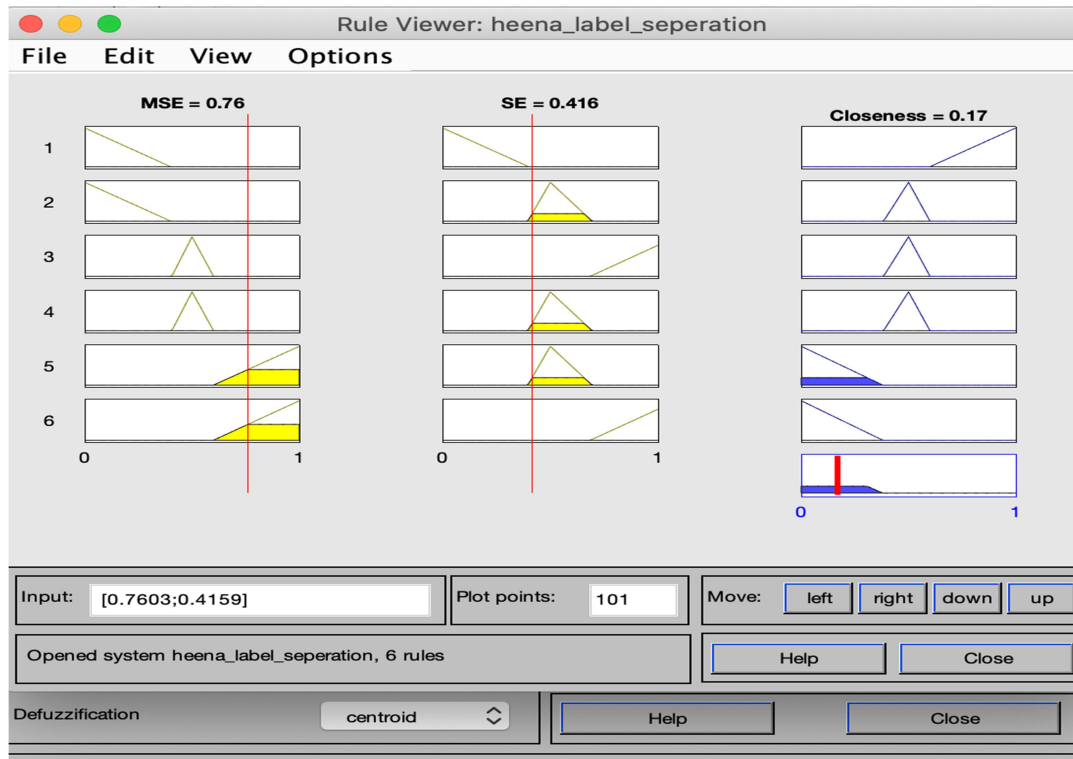


Figure 5.13 High MSE and Moderate SE Will Result into Low Closeness

The closeness is a measure of finding the relativity between the row attributes with other data row attributes. The proposed algorithm architecture illustrates 3 different attacks as explained earlier in this section and hence the ordinal measures of these types of security threats are as follows. The surface view of the same architecture is shown in 5.13.



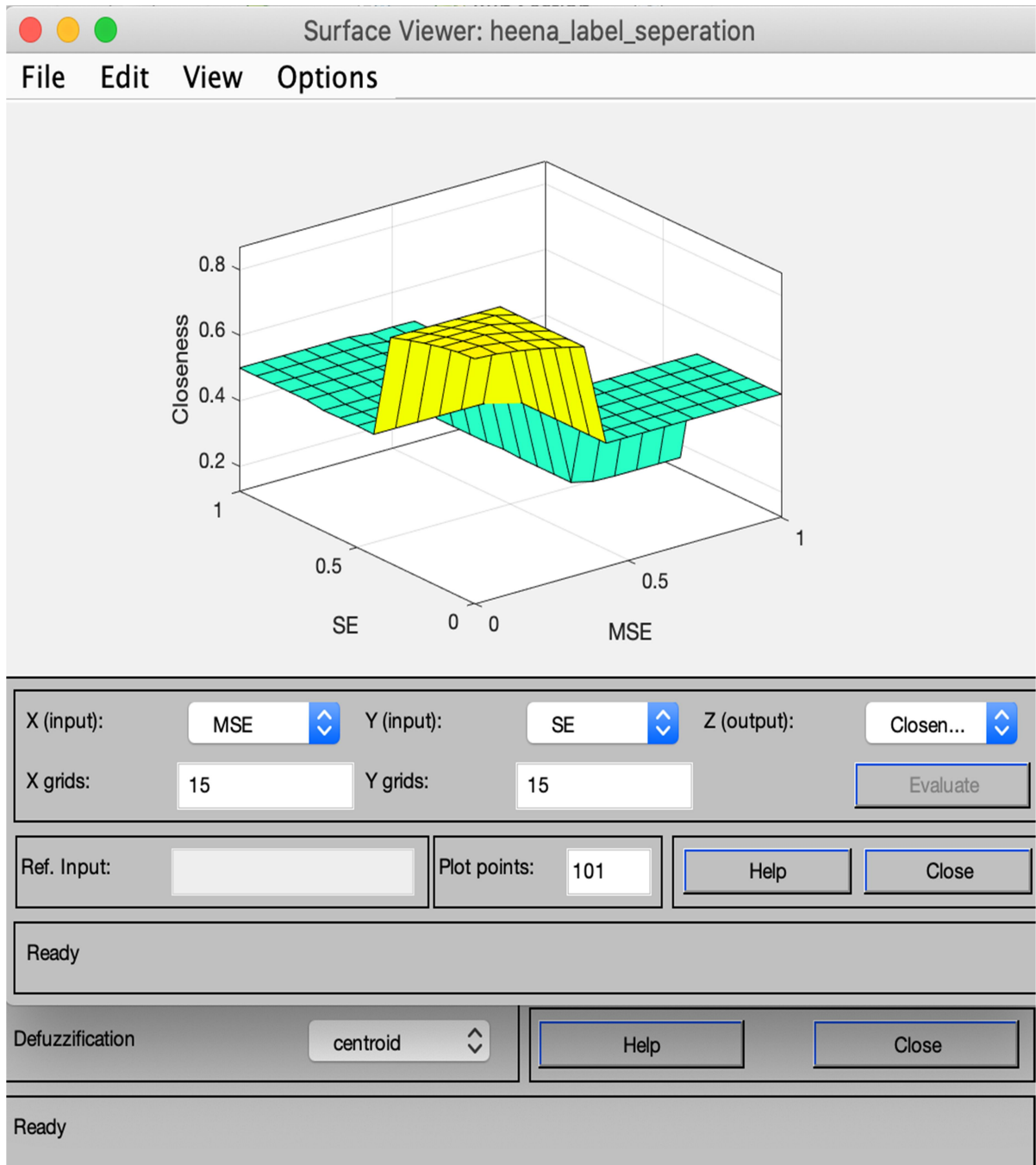


Figure 5.14 The Surface View of the Inference Engine

Combining the architecture of Mamdani and Sugeno, the proposed work utilized a hybrid value of both trapezoidal and triangular membership functions by adding additional evaluation criteria. The outcome is selected as the average value of the mapping of both triangular and trapezoidal values.

1. Fis1 = addMF(fis, "MSE – SE", "traiangular", [0.0 0.33], "Name", "GOOD");
2. Fis1 =  
    addMF(fis, "MSE – SE", " traiangular ", [.33 .50], "Name", "Moderate");
3. Fis1 = addMF(fis, "MSE – SE", " traiangular ", [.50 1], "Name", "Bad");
4. Fis2 = addMF(fis, "MSE – SE", " trapezoidal ", [0.0 0.33], "Name", "Good");
5. Fis2 =  
    addMF(fis, "MSE – SE", " trapezoidal ", [.33 .50], "Name", "Moderate");
6. Fis2 = addMF(fis, "MSE – SE", " trapezoidal ", [.50 1], "Name", "Bad");
7. Fis.Add((Fis1. +Fis2)/2)

Based on the developed inference value, the inference system executes the addition instruction set illustrated as follows.

1. fis = and output(fis, [SE, MSE], "Name", "ClassLabel");
2. fis = addMF(fis, "cl", "{trimf, trapezoidal}", [0 5 10], "Name", "Good");
3. fis = addMF(fis, "cl", "{trimf, trapezoidal} [.3, .5], "Name", "Moderate");
4. fis = addMF(fis, "cl", "", "{trimf, trapezoidal}", [.50 1], "Name", "Bad");

The hybrid fuzzy system provides more relevance toward the judgment of the class category and reduces the overall computation complexity that contributes to lowering the overall jitter in the network.

The lowest closeness refers to DDoS Class, moderate one would signify Replay and the highest would be categorized as others.

The separated dataset architecture is passed to the Neural Network utilizing the following ordinal measures that are illustrated in Table 5.2. The proposed work has also utilized other ML algorithms to validate the proposed work.

Table 5.2 Ordinals of Neural Network

|                            |              |
|----------------------------|--------------|
| Propagation Architecture   | Neurological |
| Neuron Count               | 5-25         |
| Nature of propagation      | Progressive  |
| Propagation behavior model | Levenberg    |

|                       |  |
|-----------------------|--|
| Root node validation  | Mean Squared Error (MSE)   |
| Validation parameters | a) Total number of epochs<br>b) Gradient<br>c) Count of fails in the validation  |
| Cross-validation      | Linear Regression  |
| Regression equation   | $z = ax + b$ (eq hh)<br>Where x is a multi-objective fitness function defined by a sigmoid function of neural networks |

### 5.3 Artificial Neural Network (ANN)

The artificial neural network gets its name from the working of neurons in the brain. A neural network is a system composed of computing units-artificial neurons function similar to neurons in a biological brain. Like biology, artificial neurons receive and process information, and then transmit it further. By interacting with each other, neurons can solve complex problems, including:

- Object class definition,
- Identify dependencies and aggregate data,
- Divide the received data into several groups according to the specified characteristics,
- Forecast etc.

Neurons are special biological cells that process information as shown in Figure 5.15. It consists of cell bodies or somatic cells and two types of external dendritic branches: axons and dendrites. The cell body includes a nucleus and plasma. The nucleus contains information about genetic characteristics, while plasma contains molecular information to produce materials needed for neurons. A neuron receives signals in the form of pulses from other neurons through the dendrites (receivers) and transmits signals generated by the cell body along the axons (transmitters), which passes into strands at the ends known as synapses.

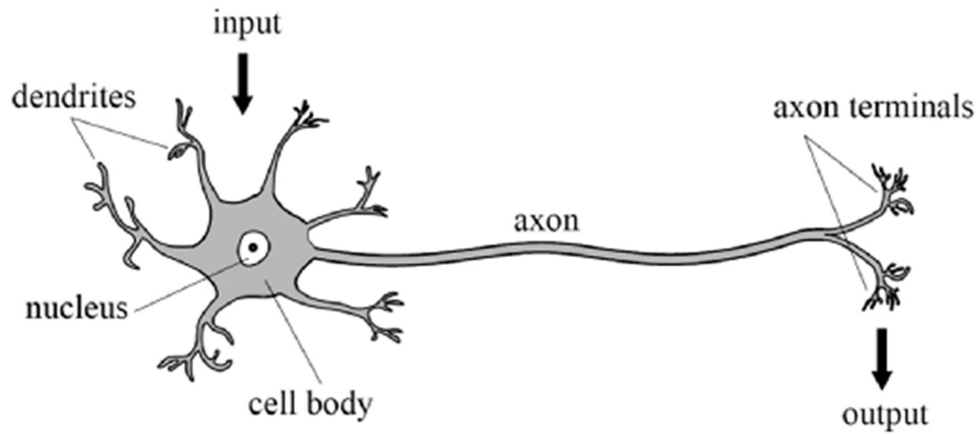


Figure 5.15 Biological Neuron

### 5.3.1 Working of ANN

Learning ability is the basic attribute of the brain. In the context of artificial neural networks, the learning process can be viewed as establishing network architecture and link weights to perform special tasks effectively. Usually, the neural network must adjust the link weights of the available training samples. As the weights are continuously adjusted, network performance will also improve.

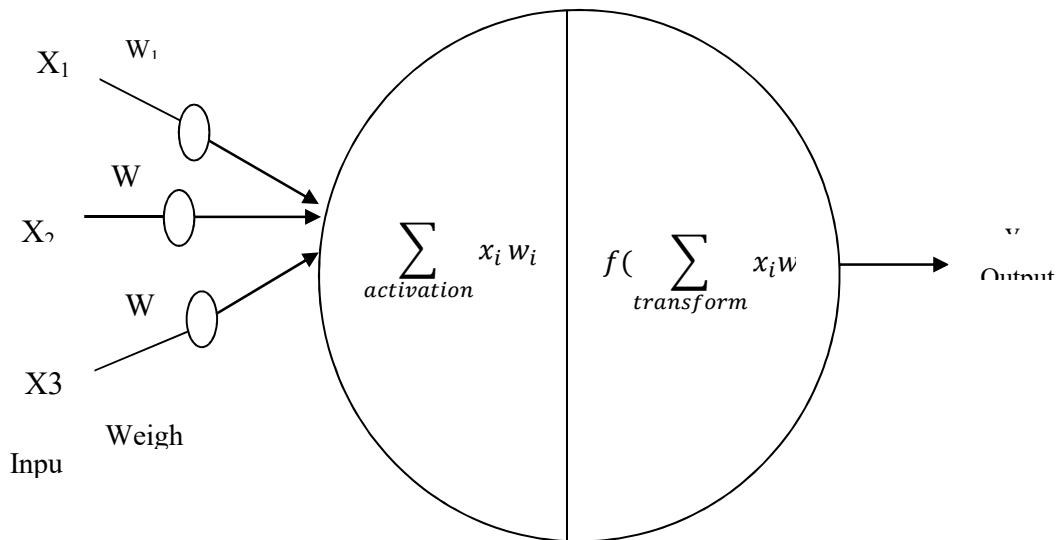


Figure 5.16 Structure of ANN

Artificial Neural Networks (ANNs) are biologically inspired computer programs designed to process information in the same manner as the human brain. ANN detects patterns and data relationships and learns through experience rather than collecting knowledge from programming. ANN consists of hundreds of units. These artificial neurons are also called processing elements, which are related to weights and create neural structures and arrange them in layers as shown in Figure 5.16.

### 5.3.2 Implementation Architecture of ANN in MATLAB

The implementation of ANN has been done utilizing the Neural Network toolbox that is supplied by the MathWorks community to support the researchers. The proposed implementation architecture is presented in the proceeding figures.

Here is the sequence of steps for the implementation of ANN through MATLAB:

1. **Data Preparation:** Start by organizing and pre-processing your training data. Ensure that the data is properly formatted and normalized, as appropriate for your specific problem.
2. **Network Creation:** Use MATLAB's Neural Network Toolbox to create an ANN model. You can choose the architecture and configuration of the network, including the number of layers, neurons per layer, and activation functions.
3. **Network Training:** Use the prepared training data to train the neural network model. MATLAB provides various training algorithms, such as back propagation, to update the network weights and biases based on the input data.
4. **Performance Evaluation:** Assess the performance of your trained ANN using validation or test data. MATLAB provides functions to compute performance metrics like accuracy, mean squared error, or classification rates.
5. **Prediction or Classification:** Once the ANN is trained and evaluated, you can use it to make predictions or classify new data points based on the learned patterns.

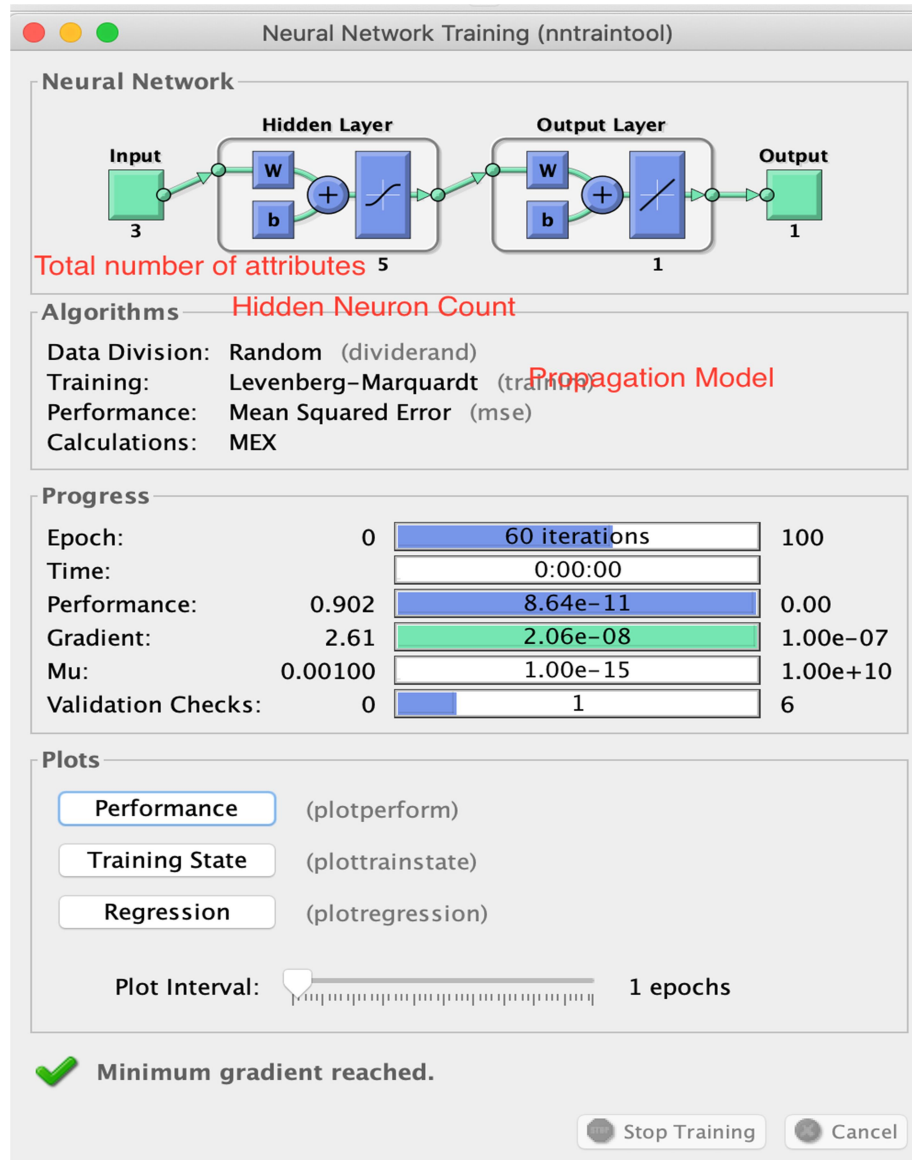


Figure 5.17 Propagation Architecture of Neural Network

To train the system, the network is passed with two elements namely the training data and the associated label which is generated in the earlier section of this research. Figure 5.17 represents a sample of the training model attained by ANN. The proposed model uses two hidden layers & 5-10 neurons per layer

ANN supports a lot of architectural training support like generation of the gradient and performing regression over multiple attribute sets via three regression measures namely, the R-value of training data, the R-value of overall test data, and the R-value of the

validation data. The overall regression value is calculated by summing up all the R values.

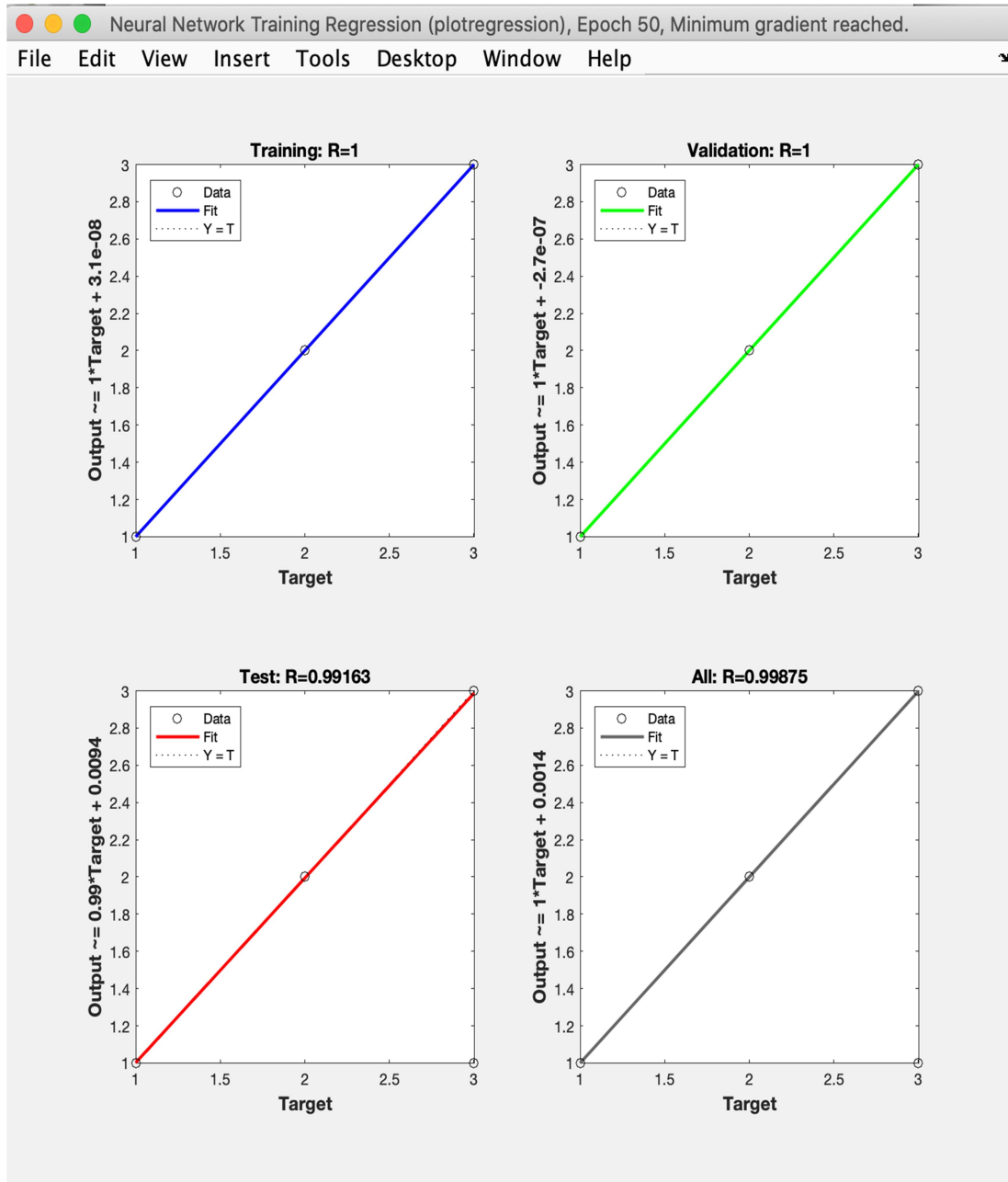


Figure 5.18 Neural Regression Architecture

As shown in Figure 5.18, the regression value has been evaluated for training, validation, and test data and the final regression value is the arithmetic mean of all three regression

values. As the neural network is a propagation-based architecture the R-value changes even for the same set of input values, as shown in Figure 5.19.

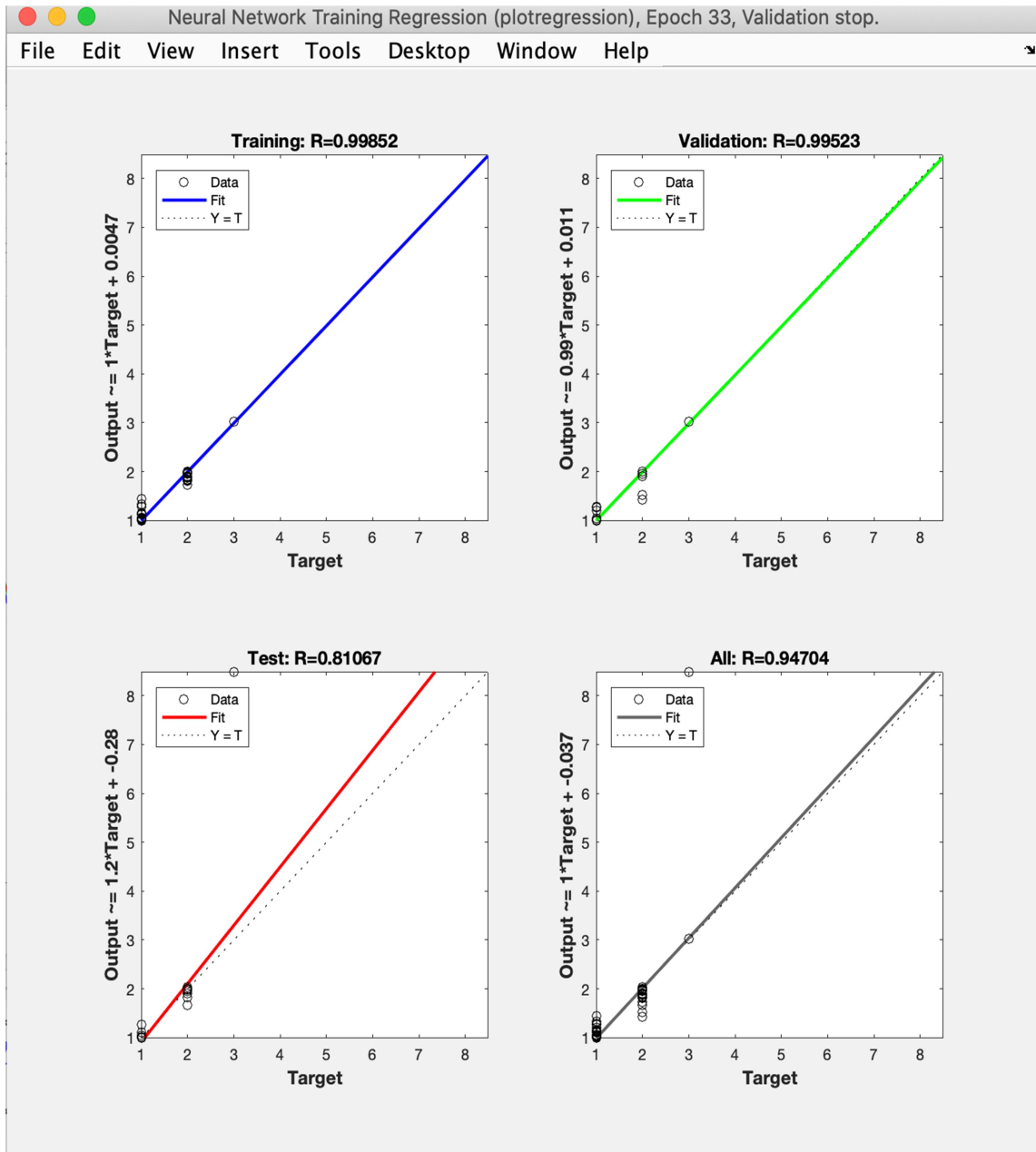


Figure 5.19 Regression Change Behaviour

The cross-validation of the utilized Neural Network is always performed using gradient and MSE analysis as shown in Figures 5.19 and 5.20.



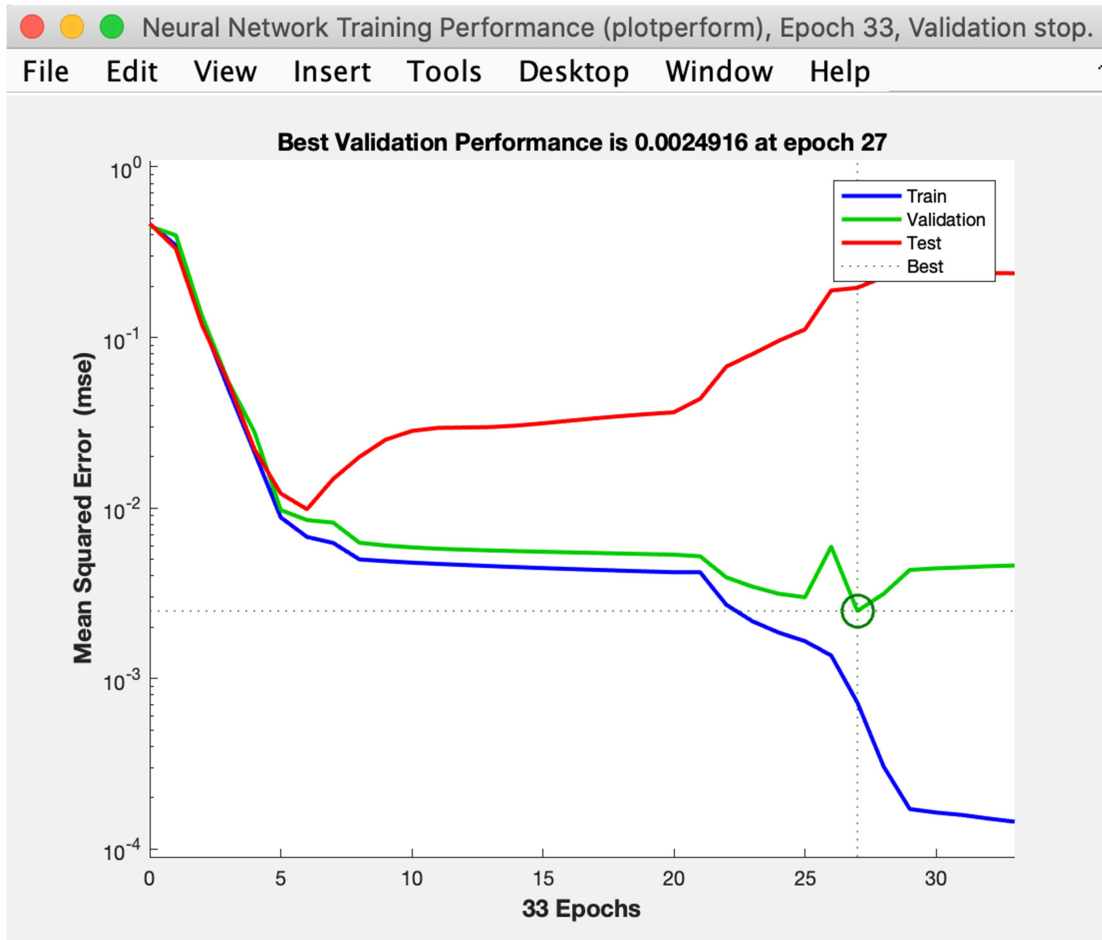
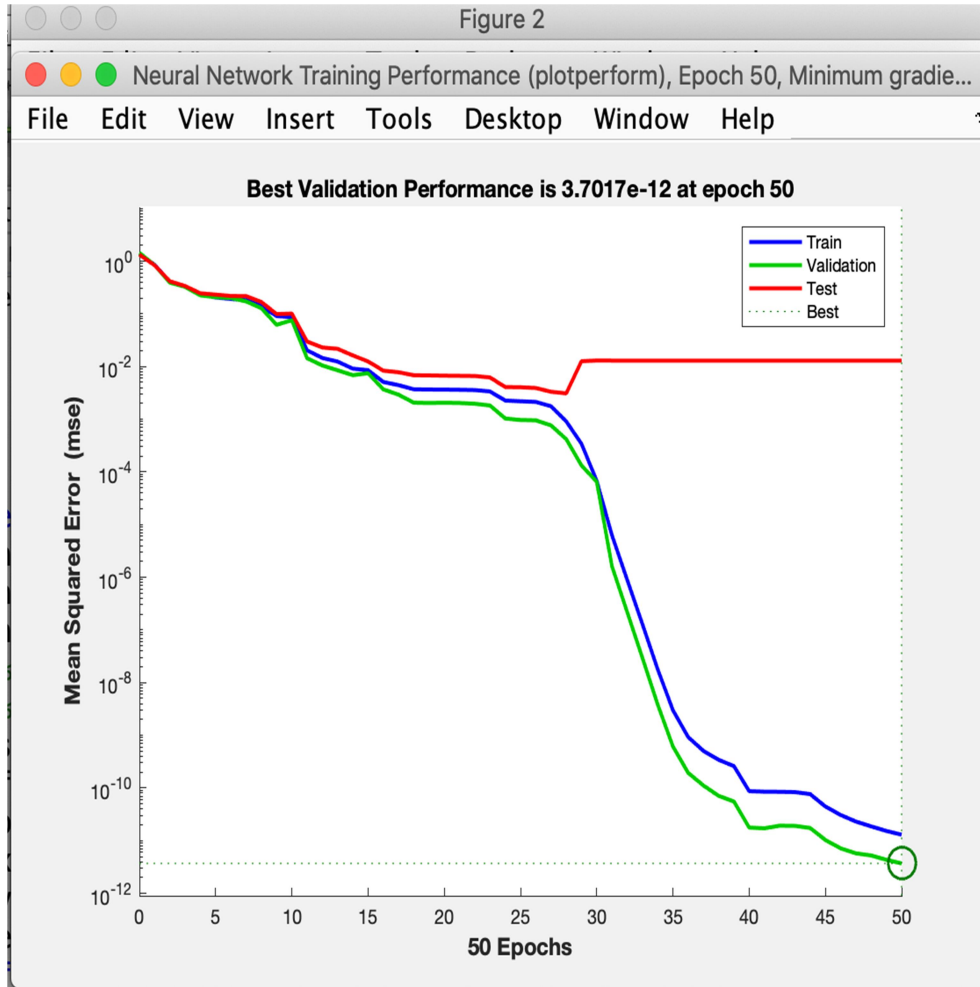


Figure 5.20 MSE Validation Scenario 1

The previous analysis has been done for the same simulation data; still, the variance in the MSE range and validation epochs is the result of the convergence of the Neural Network which supports a random walk selection model for propagation.



**Figure 5.21 MSE for Scenario 2**

The trained system is stored in the repository and when the simulation takes place if the network administrator observes that the packet flow architecture is labelled as DDoS, it will indicate to stop the data transfer and a heavy amount of battery drainage will be saved which will result into more throughput and improved delivery ratio.

Let us consider a route in  $\{55,32,21,56,17\}$  where 55 is the source node, 17 is the terminal node and 32,21 and 56 are respective heads of the communication. The proposed work ranks out 32,21 and 56 based on the classification score. Let us assume that the route is classified as DDoS and the classification score is 90%. The classification score is calculated as the relevance of the classified class to the original class. Each member viz. 32,21 and 56 shares equal score and hold (-)30% each. The negative sign is due to the classification under the attack category viz. DDoS. Higher the score, safer is the root. The

score is stored in the repository and at the time of route formation, the asking node also checks the rank of the responding node despite of only looking for least computation cost.

#### 5.4 Effect of Training Algorithm Based on Quantitative Parameters

As the proposed algorithm has used training and classification architecture, the evaluation of this section has been made based on the quantitative parameters viz. true positive rate (TPR), false positive rate (FPR), and overall classification accuracy. The proposed algorithm architecture has been compared with other state of art classification algorithms namely, Naïve Bayes and Random Forest. The naïve Bayes is an algorithm that is capable of resolving binary as well as multiclass problems. In contrast to this random forest is a machine learning algorithm capable of performing both regression and classification. It is an ensemble of a large number of minute decision trees also labelled as estimators during simulation analysis and are kept 100 as a default value.

This section has demonstrated both the supervised and semi-supervised learning results as described in the introduction section of the dissertation. For the application of the semi-supervised approach, a training ratio of 70-30, 80-20, and 90-10 has been kept and evaluation parameters viz TPR, FPR, Class Accuracy, and Confusion Matrix of the class accuracy have been analysed.

##### 5.4.1 Evaluation Using 70:30 Distribution

Table 5.3 demonstrates the TPR and FPR for 70:30 divisions and the results are compared with the existing techniques such as Naïve Bayes and Random Forest. The average value for TPR and FPR using the proposed technique is 0.96 and 0.065 respectively while the existing technique shows a TPR of about 0.93 for both Naïve Bayes and 0.94 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.084 and 0.08 respectively. Thus, the proposed technique shows prominent FPR and TPR in comparison to Naïve Bayes and Random Forest.

Table 5.3 Performance Analysis using 70:30 Data Division

| Total number of simulation records for training | TPR with Naïve Bayes | TPR with Naïve Bayes Gaussian | TPR with Random Forest | Proposed FPR | FPR with Naïve Bayes | FPR with Naïve Bayes Gaussian | FPR with Random Forest |
|---|----------------------|-------------------------------|------------------------|--------------|----------------------|-------------------------------|------------------------|
| 1000  | 0.899323             | 0.909037                      | 0.915635               | 0.105441     | 0.130677             | 0.120963                      | 0.114365               |
| 2000  | 0.902093             | 0.905411                      | 0.905874               | 0.104143     | 0.127908             | 0.124589                      | 0.124126               |
| 3000  | 0.902334             | 0.902845                      | 0.904905               | 0.100806     | 0.127666             | 0.127155                      | 0.125095               |
| 4000  | 0.913379             | 0.922129                      | 0.930408               | 0.090424     | 0.11663              | 0.107871                      | 0.099592               |
| 5000  | 0.918613             | 0.920255                      | 0.924255               | 0.084707     | 0.111387             | 0.109745                      | 0.105745               |
| 6000  | 0.929637             | 0.938062                      | 0.948043               | 0.075522     | 0.100363             | 0.091938                      | 0.081957               |
| 7000  | 0.925978             | 0.935206                      | 0.936987               | 0.073274     | 0.104022             | 0.094794                      | 0.093013               |
| 8000  | 0.935326             | 0.943593                      | 0.945831               | 0.069824     | 0.094674             | 0.086407                      | 0.084169               |
| 9000  | 0.948432             | 0.954675                      | 0.964559               | 0.059586     | 0.081568             | 0.075326                      | 0.065441               |
| 10000   | 0.970211             | 0.979666                      | 0.983903               | 0.03778      | 0.059789             | 0.050334                      | 0.046097               |
| 20000   | 0.976188             | 0.9791                        | 0.980164               | 0.029746     | 0.053812             | 0.0509                        | 0.049836               |
| 50000   | 0.974655             | 0.982345                      | 0.984171               | 0.02848      | 0.055345             | 0.047655                      | 0.045829               |
| 70000   | 0.977452             | 0.983593                      | 0.98806                | 0.026471     | 0.052548             | 0.046407                      | 0.04194                |
| 100000  | 0.978124             | 0.982251                      | 0.987472               | 0.021733     | 0.051876             | 0.047749                      | 0.042528               |
| Average   | 0.939411             | 0.945584                      | 0.950019               | 0.064853     | 0.09059              | 0.084417                      | 0.079981               |

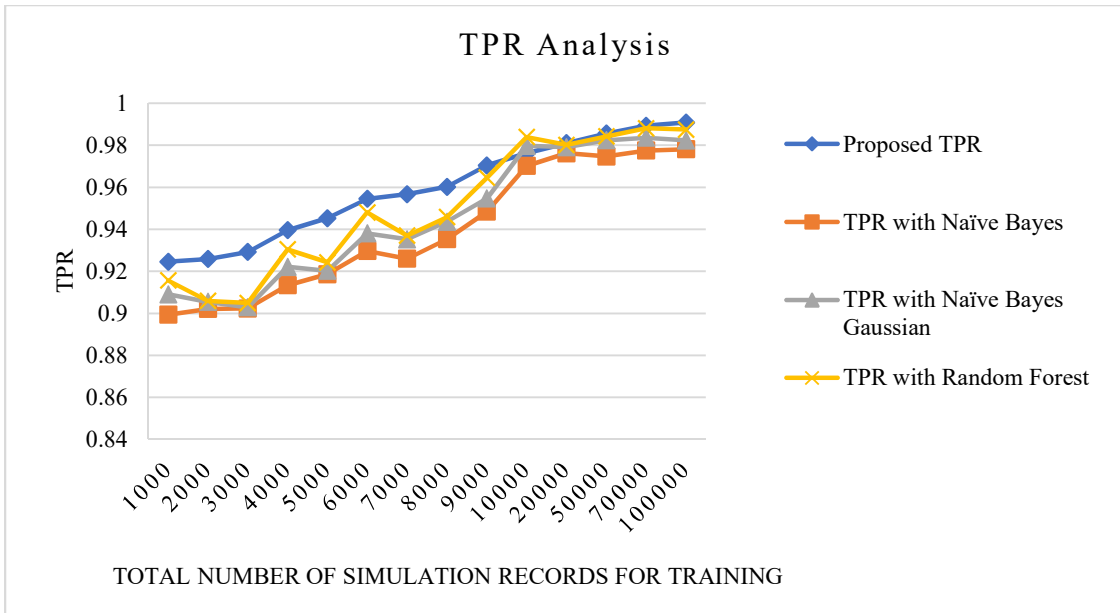


Figure 5.22 TPR Analysis using 70:30 Data Division

Figure 5.22 shows the comparative analysis of TPR using a 70-30 distribution. It is seen that as the number of simulations increases, then TPR also increases using the proposed approach. However, the TPR uses the existing techniques following the zigzag pattern and it remains constant for 7000 and 10,000 simulations.

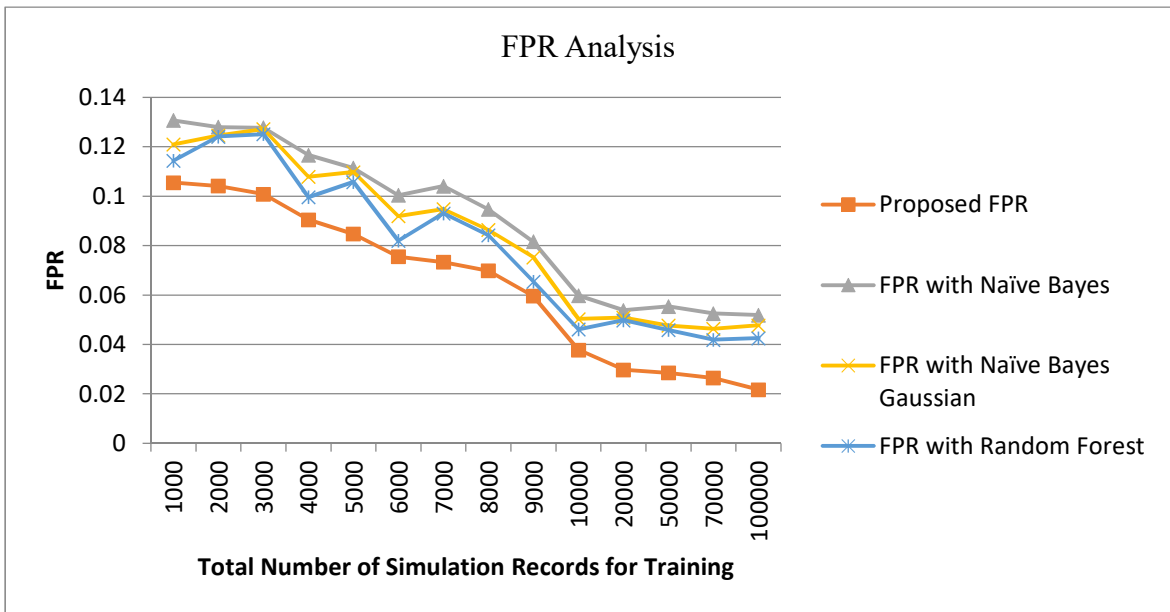


Figure 5.23 FPR Analysis using 70:30 Data Division

Figure 5.23 shows the comparative analysis of FPR using a 70-30 distribution. It is seen that as the number of simulations increases, then FPR keeps on decreasing for all the techniques. But, the proposed approach decreases exponentially. However, the FPR uses the existing techniques following the zigzag pattern and it keeps on decreasing and then remaining constant for 7000 and 10,000 simulations.

Table 5.4 Confusion Matrix for 70:30 Data Division

|        | DDOS  | REPLAY | Others | Grand Total | Total Number of Records |
|--------|-------|--------|--------|-------------|-------------------------|
| DDOS   | 54000 | 3050   | 1500   | 58550       | 100000                  |
| REPLAY | 1550  | 32100  | 2122   | 35772       |                         |
| OTHERS | 66    | 12     | 5600   | 5678        |                         |

Class accuracy distribution was observed using 70% of the data for training and 30% of the total data for testing as shown in table 5.4. It is observed that out of the total records, 58500 records were representing a DDoS attack with high-class accuracy for DDoS shown by 54000 records.

#### 5.4.2 Evaluation Using 80:20 Distribution

Table 5.5 demonstrates the TPR and FPR for 80:20 division and the results are compared with the existing techniques such as Naïve Bayes and Random Forest. The average value for TPR and FPR using the proposed technique is about 0.96 and 0.065 respectively while the existing technique shows a TPR of about 0.93 for both Naïve Bayes and 0.94 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.09 and 0.08 respectively. Thus, the proposed technique shows prominent FPR and TPR in comparison to Naïve Bayes and Random Forest.

Table 5.5 Performance Analysis using 80:20 Data Division

| Total number of | Proposed TPR | TPR with Naïve | TPR with Naïve | TPR with Rando | Proposed FPR | FPR with Naïve Bayes | FPR with Naïve Bayes | FPR with Rando |
|-----------------|--------------|----------------|----------------|----------------|--------------|----------------------|----------------------|----------------|
|                 |              |                |                |                |              |                      |                      |                |

| simulation records for training |                 | Bayes        | Bayes Gaussian | m Forest     |              |          | Gaussian | m Forest     |
|---------------------------------|-----------------|--------------|----------------|--------------|--------------|----------|----------|--------------|
| 1000                            | 0.9245<br>59    | 0.899<br>323 | 0.909037       | 0.9156<br>35 | 0.1054<br>41 | 0.130677 | 0.120963 | 0.114<br>365 |
| 2000                            | 0.9258<br>57    | 0.902<br>093 | 0.905411       | 0.9058<br>74 | 0.1041<br>43 | 0.127908 | 0.124589 | 0.124<br>126 |
| 3000                            | 0.9291<br>94    | 0.902<br>334 | 0.902845       | 0.9049<br>05 | 0.1008<br>06 | 0.127666 | 0.127155 | 0.125<br>095 |
| 4000                            | 0.9395<br>76    | 0.913<br>37  | 0.922129       | 0.9304<br>08 | 0.0904<br>24 | 0.11663  | 0.107871 | 0.099<br>592 |
| 5000                            | 0.9452<br>93    | 0.918<br>613 | 0.920255       | 0.9242<br>55 | 0.0847<br>07 | 0.111387 | 0.109745 | 0.105<br>745 |
| 6000                            | 0.9544<br>78    | 0.929<br>637 | 0.938062       | 0.9480<br>43 | 0.0755<br>22 | 0.100363 | 0.091938 | 0.081<br>957 |
| 7000                            | 0.9567<br>26    | 0.925<br>978 | 0.935206       | 0.9369<br>87 | 0.0732<br>74 | 0.104022 | 0.094794 | 0.093<br>013 |
| 8000                            | 0.9601<br>76    | 0.935<br>326 | 0.943593       | 0.9458<br>31 | 0.0698<br>24 | 0.094674 | 0.086407 | 0.084<br>169 |
| 9000                            | 0.9704<br>15    | 0.948<br>432 | 0.954675       | 0.9645<br>59 | 0.0595<br>86 | 0.081568 | 0.075326 | 0.065<br>441 |
| 10000                           | 0.9792<br>22    | 0.970<br>211 | 0.979666       | 0.9839<br>03 | 0.0377<br>8  | 0.059789 | 0.050334 | 0.046<br>097 |
| 20000                           | 0.9810<br>25    | 0.976<br>188 | 0.9791         | 0.9801<br>64 | 0.0297<br>46 | 0.053812 | 0.0509   | 0.049<br>836 |
| 50000                           | 0.9885<br>21    | 0.974<br>655 | 0.982345       | 0.9841<br>71 | 0.0284<br>8  | 0.055345 | 0.047655 | 0.045<br>829 |
| 70000                           | 0.9912<br>59    | 0.977<br>452 | 0.983593       | 0.9880<br>6  | 0.0264<br>71 | 0.052548 | 0.046407 | 0.041<br>94  |
| 100000                          | 0.9938<br>267   | 0.978<br>124 | 0.982251       | 0.9874<br>72 | 0.0217<br>33 | 0.051876 | 0.047749 | 0.042<br>528 |
| Average                         | 0.9600<br>09121 | 0.939<br>41  | 0.945584       | 0.9500<br>19 | 0.0648<br>53 | 0.09059  | 0.084417 | 0.079<br>981 |

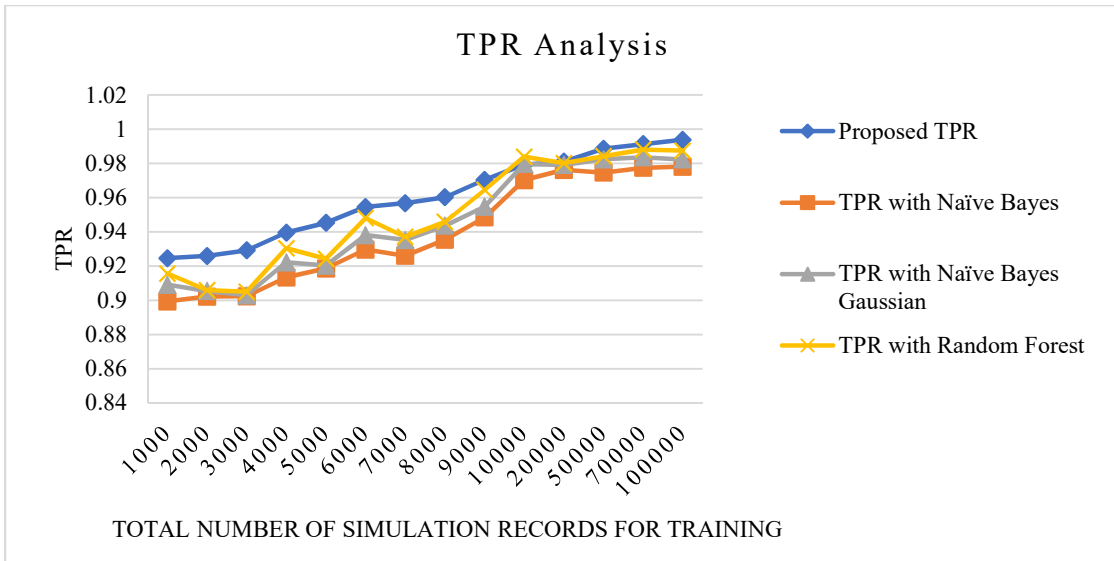


Figure 5.24 TPR Analysis using 80:20 Data Division

Figure 5.24 shows the comparative analysis of TPR using an 80-20 distribution. It is seen that as the number of simulations increases, then TPR using the proposed approach also increases. However, the TPR using the existing techniques increase or decreases and thus, follows the zigzag pattern but, it remains constant for 7000 and 10,000 simulations.

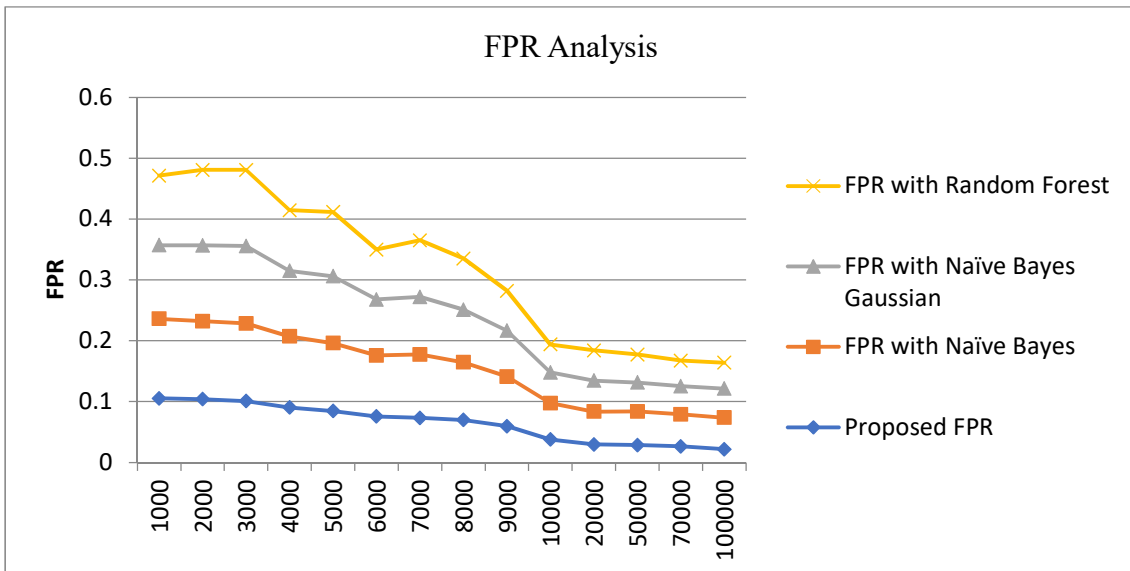


Figure 5.25 FPR Analysis using 80:20 Data Division



Figure 5.25 shows the comparative analysis of FPR using an 80-20 distribution. It is seen that as the number of simulations increases, then FPR also decreases using the proposed approach. However, the FPR uses the existing techniques following the zigzag pattern following the decreasing pattern but, it remains constant for 7000 and 10,000 simulations.

Table 5.6 Confusion Matrix for 80:20 Data Division

|        | DDOS  | REPLAY | Others | Grand Total | Total Number of Records |
|--------|-------|--------|--------|-------------|-------------------------|
| DDOS   | 56500 | 1350   | 1300   | 59150       | 100000                  |
| REPLAY | 1300  | 31200  | 2050   | 34550       |                         |
| OTHERS | 100   | 29     | 6171   | 6300        |                         |

Table 5.6 shows the Class accuracy distribution observed using 80% of the data for training and 20% of the total data for testing is shown in the given table. It is observed that out of the total records, 59150 records were representing DDoS attacks with high-class accuracy while others show accuracy for 6300 records.

### 5.4.3 Evaluation Using 90:10 Distribution

Table 5.7 demonstrates the TPR and FPR for 90:10 division and the results are compared with the existing techniques such as Naïve Bayes and Random Forest. The average value for TPR and FPR using the proposed technique is about 0.97 and 0.053 respectively while the existing technique shows TPR of about 0.95 for both Naïve Bayes and 0.958 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.07 and 0.071, respectively. Thus, the proposed technique shows prominent FPR and TPR in comparison to Naïve Bayes and Random Forest.

Table 5.7 Performance Analysis using 90:10 Data Division

| Total number of simulation records | Proposed TPR | TPR with Naïve Bayes | TPR with Naïve Bayes Gaussian | TPR with Random Forest | Proposed FPR | FPR with Naïve Bayes | FPR with Naïve Bayes Gaussian | FPR with Random Forest |
|------------------------------------|--------------|----------------------|-------------------------------|------------------------|--------------|----------------------|-------------------------------|------------------------|
|                                    |              |                      |                               |                        |              |                      |                               |                        |

|              |             |          |          |          |         |          |          |            |
|--------------|-------------|----------|----------|----------|---------|----------|----------|------------|
| for training |             |          |          |          |         |          |          |            |
| 1000         | 0.9282257   | 0.90462  | 0.90883  | 0.91136  | 0.10177 | 0.12538  | 0.12117  | 0.1186354  |
| 2000         | 0.9349928   | 0.91425  | 0.92221  | 0.92379  | 0.09501 | 0.11575  | 0.10779  | 0.1062136  |
| 3000         | 0.9566949   | 0.92818  | 0.93267  | 0.9328   | 0.07331 | 0.10182  | 0.09734  | 0.0972011  |
| 4000         | 0.9601763   | 0.92965  | 0.93622  | 0.93848  | 0.06982 | 0.10035  | 0.09378  | 0.0915155  |
| 5000         | 0.9625865   | 0.9406   | 0.94634  | 0.95327  | 0.06741 | 0.0894   | 0.08366  | 0.076735   |
| 6000         | 0.9660267   | 0.93617  | 0.93754  | 0.93915  | 0.06397 | 0.09383  | 0.09246  | 0.090846   |
| 7000         | 0.9704969   | 0.94795  | 0.95438  | 0.95511  | 0.0595  | 0.08205  | 0.07562  | 0.0748913  |
| 8000         | 0.9762958   | 0.95023  | 0.95214  | 0.95535  | 0.0537  | 0.07977  | 0.07786  | 0.0746544  |
| 9000         | 0.9796536   | 0.95506  | 0.96384  | 0.96666  | 0.05035 | 0.07494  | 0.06616  | 0.063345   |
| 10000        | 0.988697    | 0.95893  | 0.95913  | 0.96417  | 0.0413  | 0.07107  | 0.07087  | 0.0658273  |
| 20000        | 0.98786     | 0.98154  | 0.98654  | 0.98889  | 0.02403 | 0.04846  | 0.04346  | 0.0411073  |
| 50000        | 0.997546    | 0.98324  | 0.99231  | 0.9999   | 0.01947 | 0.04676  | 0.03769  | 0.0300966  |
| 70000        | 0.99874     | 0.98733  | 0.9901   | 0.99931  | 0.01597 | 0.04267  | 0.0399   | 0.030694   |
| 100000       | 0.99541     | 0.98922  | 0.99073  | 0.99364  | 0.01261 | 0.04078  | 0.03927  | 0.036359   |
| Average      | 0.976559921 | 0.950497 | 0.955212 | 0.958706 | 0.05344 | 0.079503 | 0.074788 | 0.07129439 |

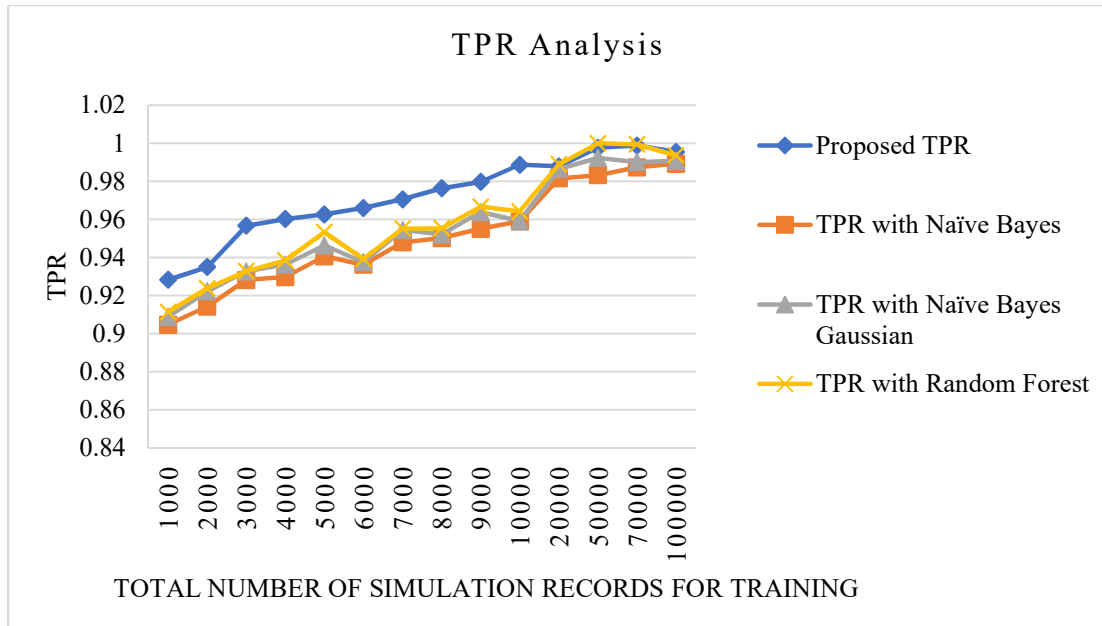


Figure 5.26 TPR Analysis using 90:10 Data Division

Figure 5.26 shows the comparative analysis of TPR using a 90-10 distribution. It is seen that as the number of simulations increases, then TPR using the proposed approach also increases and remains constant in increasing order. However, the TPR using the existing techniques keeps on increasing and decreasing. The proposed technique shows prominent results in comparison to the existing technique.

Figure 5.27 shows the comparative analysis of FPR using a 90-10 distribution. It is seen that as the number of simulations increases, the FPR keeps on decreasing for all the techniques. But, the proposed approach decreases exponentially. However, the FPR using the existing techniques keeps on increasing and decreasing and thus, increases for 7000 and 10,000 simulations.

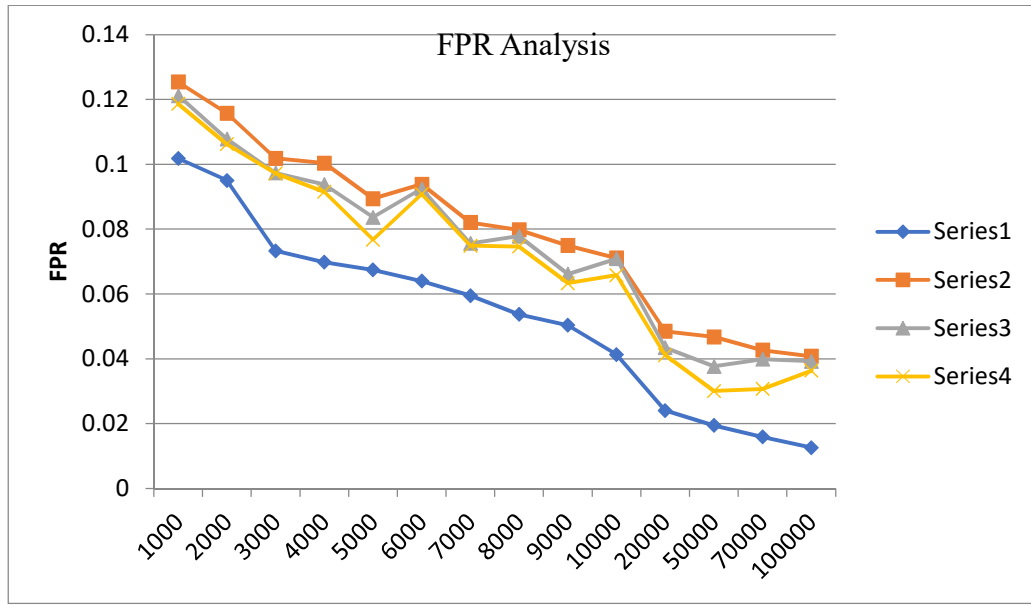


Figure 5.27 FPR Analysis using 90:10 Data Division

Table 5.8 Confusion Matrix for 90:10 Data Division

|        | DDOS  | REPLAY | Others | Grand Total | Total Number of Records |
|--------|-------|--------|--------|-------------|-------------------------|
| DDOS   | 57500 | 1000   | 1200   | 59700       | 100000                  |
| REPLAY | 1300  | 31200  | 1500   | 34000       |                         |
| OTHERS | 75    | 50     | 6175   | 6300        |                         |

Table 5.8 shows the Class accuracy distribution observed using 90% of the data for training and 10% of the total data for testing is shown in the given table. It is observed that out of the total records, 59700 records were representing DDoS attacks with high-class accuracy while others show accuracy for 6300 records.

### 5.5 Summary of the Chapter

This chapter discusses the different Machine learning algorithms such as SVM in three different categories. The classifier detail also elaborated with the working of ANN. The implementation of ANN in MATLAB, Neural Regression architecture, and regression

analysis are also detailed well. The MSE validation behaviour for different scenarios is also presented in this chapter.

## Chapter 6: Results and Comparison

---

The chapter presents the result and outcomes of the detailed simulation analysis performed to justify the effectiveness of the proposed work. The evaluation is first performed against different classifiers in terms of QoS parameters. This is then followed by a comparative analysis of the existing studies.

### 6.1 Background Information

The proposed work has been evaluated considering the effect of training and classification algorithms on QoS parameters. As the objective suggests itself that this chapter will illustrate the effect on QoS parameters based on the proposed architecture system. Hence, this portion will describe the architecture and evaluation of Throughput, PDR, TDR, and Jitter

### 6.2 QoS Effect Based on Different Classifiers

Table 6.1 depicts the throughput results using the shared approach and the already existing algorithms. It is seen that the average throughput using the proposed approach is 19219.39 Kbps while throughput with Naïve Bayes Gaussian is 13212.89 kbps. Consequently, throughput using the Random Forest is 13014.08 kbps. Thus, throughput using the proposed technique is much better than the other techniques.

Table 6.1 Evaluation of Throughput Based on Different Classifiers

| Packet Injection Rate (Packets/sec) | Total number of Simulations | Throughput Proposed with FFBPNN Algorithm | Throughput with Naïve Bayes | Throughput with Naïve Bayes Gaussian | Throughput with Random Forest |
|-------------------------------------|-----------------------------|---|-----------------------------|--------------------------------------|-------------------------------|
| 10000                               | 1000                        | 9489.7536                                 | 10098.4896                  | 9108.4776                            | 9590.0196                     |
| 10000                               | 2000                        | 10111.0254                                | 9725.9142                   | 10072.7652                           | 8376.5154                     |
| 10000                               | 5000                        | 9728.709                                  | 8707.4952                   | 9056.9064                            | 8430.7386                     |
| 10000                               | 10000                       | 9745.131                                  | 8992.8096                   | 9441.8442                            | 8114.7936                     |

|       |        |           |           |           |           |
|-------|--------|-----------|-----------|-----------|-----------|
| 10000 | 20000  | 9902.721  | 9106.2234 | 9246.0858 | 7871.697  |
| 10000 | 50000  | 9278.3382 | 9648.486  | 8097.2088 | 8452.2402 |
| 10000 | 100000 | 9347.6574 | 8822.8164 | 8294.283  | 7711.761  |
| 15000 | 1000   | 13831.098 | 15063.156 | 11780.082 | 14203.602 |
| 15000 | 2000   | 14399.85  | 13518.468 | 13429.32  | 12353.424 |
| 15000 | 5000   | 14192.586 | 13944.624 | 12752.856 | 13125.768 |
| 15000 | 10000  | 14127.102 | 14433.714 | 12982.968 | 12643.716 |
| 15000 | 20000  | 14709.216 | 15267.156 | 12631.986 | 13161.06  |
| 15000 | 50000  | 14905.974 | 14274.696 | 13693.194 | 14064.27  |
| 15000 | 100000 | 14683.614 | 15132.516 | 13127.196 | 13236.438 |
| 20000 | 1000   | 19989.654 | 17731.068 | 17066.538 | 16487.586 |
| 20000 | 2000   | 20124.906 | 19949.364 | 17565.93  | 18469.752 |
| 20000 | 5000   | 18964.35  | 17624.886 | 16450.764 | 16712.394 |
| 20000 | 10000  | 19307.886 | 18511.878 | 16535.934 | 16350.702 |
| 20000 | 20000  | 19495.158 | 18969.756 | 17575.314 | 17447.202 |
| 20000 | 50000  | 20376.948 | 19831.656 | 19861.848 | 19372.044 |
| 20000 | 100000 | 18761.064 | 17544.51  | 18699.15  | 17119.884 |

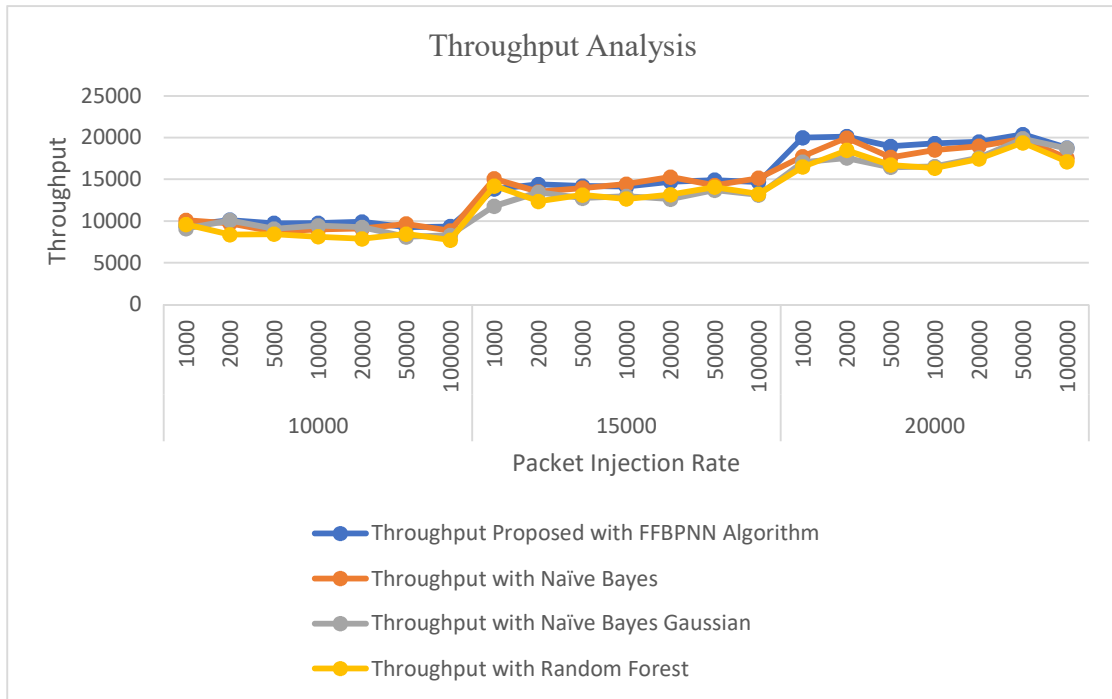


Figure 6.1 Evaluation of Throughput Based on Different Classifiers

Figure 6.1 depicts the comparative evaluation of the shared technique with the existing technique. The graphical analysis shows that the throughput using the proposed approach is virtually found to remain constant for different simulation rounds. However, some increase is observed in the throughput as the packet injection rate is increased. The results show that throughput using the proposed technique keeps on increasing with an increase in packet injection rate for all the studies with the PDR of the proposed work remaining higher among all.

Table 6.2 Evaluation of PDR Based on Different Classifiers

| Packet Injection Rate (Packets/second) | Total number of Simulations | PDR Proposed | PDR With Naïve Bayes | PDR With Naïve Bayes Gaussian | PDR With Random Forest |
|--|-----------------------------|--------------|----------------------|-------------------------------|------------------------|
| 10000                                  | 1000                        | 0.91274      | 0.88745              | 0.89574                       | 0.89874                |
| 10000                                  | 2000                        | 0.91901      | 0.89243              | 0.90247                       | 0.90668                |
| 10000                                  | 5000                        | 0.92174      | 0.89574              | 0.90351                       | 0.908473               |
| 10000                                  | 10000                       | 0.92212      | 0.89636              | 0.90417                       | 0.91025                |
| 10000                                  | 20000                       | 0.92354      | 0.89735              | 0.90485                       | 0.913851               |
| 10000                                  | 50000                       | 0.92568      | 0.8998               | 0.90517                       | 0.914261               |
| 10000                                  | 100000                      | 0.92842      | 0.89945              | 0.90598                       | 0.91558                |
| 15000                                  | 1000                        | 0.93538      | 0.91013              | 0.91992                       | 0.927914               |
| 15000                                  | 2000                        | 0.93646      | 0.91314              | 0.92194                       | 0.925487               |
| 15000                                  | 5000                        | 0.93732      | 0.91475              | 0.92422                       | 0.927782               |
| 15000                                  | 10000                       | 0.93976      | 0.91532              | 0.92584                       | 0.934575               |
| 15000                                  | 20000                       | 0.94182      | 0.91644              | 0.92314                       | 0.929312               |
| 15000                                  | 50000                       | 0.94112      | 0.91762              | 0.92412                       | 0.930801               |
| 15000                                  | 100000                      | 0.94224      | 0.91874              | 0.92547                       | 0.933167               |
| 20000                                  | 1000                        | 0.95152      | 0.92132              | 0.92654                       | 0.931138               |
| 20000                                  | 2000                        | 0.95157      | 0.92187              | 0.93214                       | 0.931382               |
| 20000                                  | 5000                        | 0.95193      | 0.92245              | 0.93481                       | 0.930434               |
| 20000                                  | 10000                       | 0.95514      | 0.92298              | 0.93813                       | 0.937904               |
| 20000                                  | 20000                       | 0.95721      | 0.92384              | 0.93974                       | 0.935976               |
| 20000                                  | 50000                       | 0.95847      | 0.92457              | 0.94314                       | 0.936578               |



|       |        |         |         |         |          |
|-------|--------|---------|---------|---------|----------|
| 20000 | 100000 | 0.96072 | 0.92587 | 0.94874 | 0.941882 |
|-------|--------|---------|---------|---------|----------|

Table 6.2 shows the PDR using the shared approach and the already existing techniques. It is seen that PDR using the proposed approach is 0.94 while PDR with Naïve Bayes Gaussian is 0.91. Further, PDR using the Random Forest is 0.93. Thus, PDR using the proposed work is higher than the other technique used in the analysis at different simulation rounds and packet injection rates.

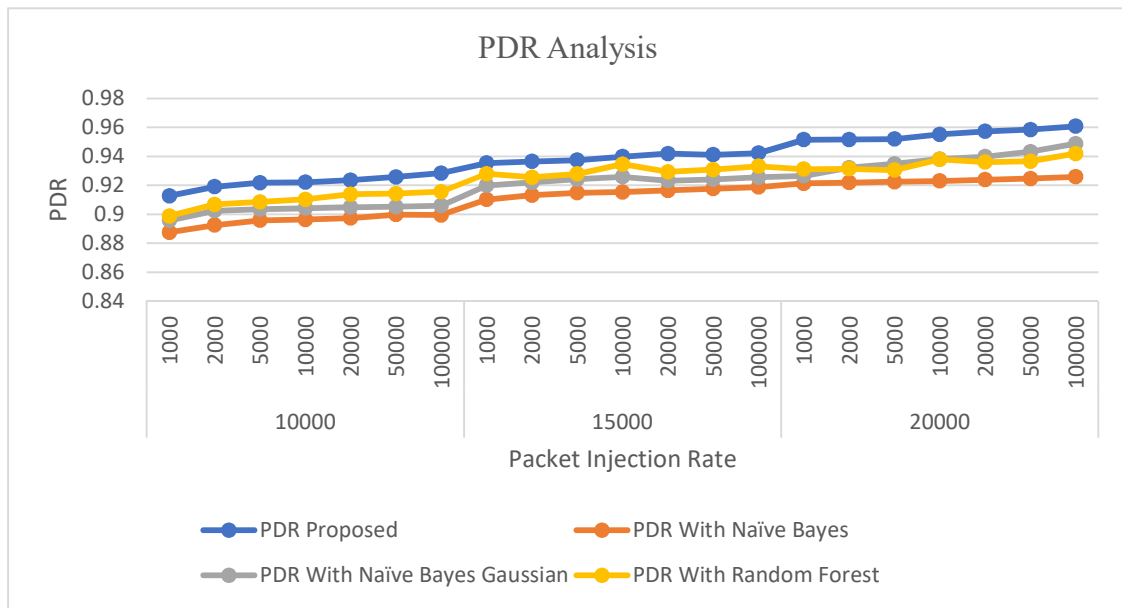


Figure 6.2 Evaluation of PDR Based on Different Classifiers

Figure 6.2 depicts the graphical representation of the PDR comparative analysis. The evaluation results show that PDR using the proposed technique remains virtually constant with an increase in the simulation rounds however, with an increase in the packet injection rate PDR of all the techniques gets increased.

Table 6.3 presents the TDR analysis of the four approaches. The average TDR exhibited by the proposed approach is 94.65% while TDR using Naïve Bayes Gaussian is 92.30% and TDR using the Random Forest is 92.98%. Thus, the parametric values show that the TDR using the proposed technique remains higher over the variation in the simulation rounds and the packet injection rate.

Table 6.3 Evaluation of TDR Based on Different Classifiers

| Packet Injection Rate (Packets/second) | Total number of Simulations | TDR Proposed with FFBPNN Algorithm | TDR with Naïve Bayes | TDR with Naïve Bayes Gaussian | TDR with Random Forest |
|--|-----------------------------|------------------------------------|----------------------|-------------------------------|------------------------|
| 10000                                  | 1000                        | 92.502576                          | 84.5376              | 86.4557                       | 89.023                 |
| 10000                                  | 2000                        | 93.521148                          | 85.1516              | 86.9037                       | 89.232                 |
| 10000                                  | 5000                        | 94.498512                          | 85.2052              | 86.927                        | 89.5775                |
| 10000                                  | 10000                       | 95.175384                          | 85.9842              | 87.5331                       | 90.0375                |
| 10000                                  | 20000                       | 95.892036                          | 86.3724              | 88.2122                       | 90.0957                |
| 10000                                  | 50000                       | 96.219456                          | 87.1325              | 88.7301                       | 90.6936                |
| 10000                                  | 100000                      | 96.407952                          | 87.795               | 89.7304                       | 90.935                 |
| 15000                                  | 1000                        | 96.472212                          | 87.86                | 90.1075                       | 90.944                 |
| 15000                                  | 2000                        | 96.550038                          | 88.6734              | 91.0431                       | 91.9037                |
| 15000                                  | 5000                        | 97.035354                          | 89.3764              | 91.8233                       | 92.2724                |
| 15000                                  | 10000                       | 97.192944                          | 89.4686              | 92.3502                       | 93.0548                |
| 15000                                  | 20000                       | 97.569018                          | 89.7567              | 92.7775                       | 93.7517                |
| 15000                                  | 50000                       | 98.587182                          | 90.2809              | 93.3226                       | 93.845                 |
| 15000                                  | 100000                      | 98.607684                          | 91.1295              | 94.3259                       | 94.0133                |
| 20000                                  | 1000                        | 98.627574                          | 91.4901              | 94.9467                       | 94.8474                |
| 20000                                  | 2000                        | 98.771394                          | 92.2509              | 95.9112                       | 95.253                 |
| 20000                                  | 5000                        | 98.885124                          | 93.1882              | 96.5126                       | 95.7255                |
| 20000                                  | 10000                       | 99.410934                          | 93.3228              | 96.9159                       | 96.0415                |
| 20000                                  | 20000                       | 99.855756                          | 93.8473              | 97.2192                       | 96.7893                |
| 20000                                  | 50000                       | 99.23454                           | 94.7079              | 98.009                        | 97.2032                |
| 20000                                  | 100000                      | 99.25112                           | 95.3327              | 98.6289                       | 97.5051                |

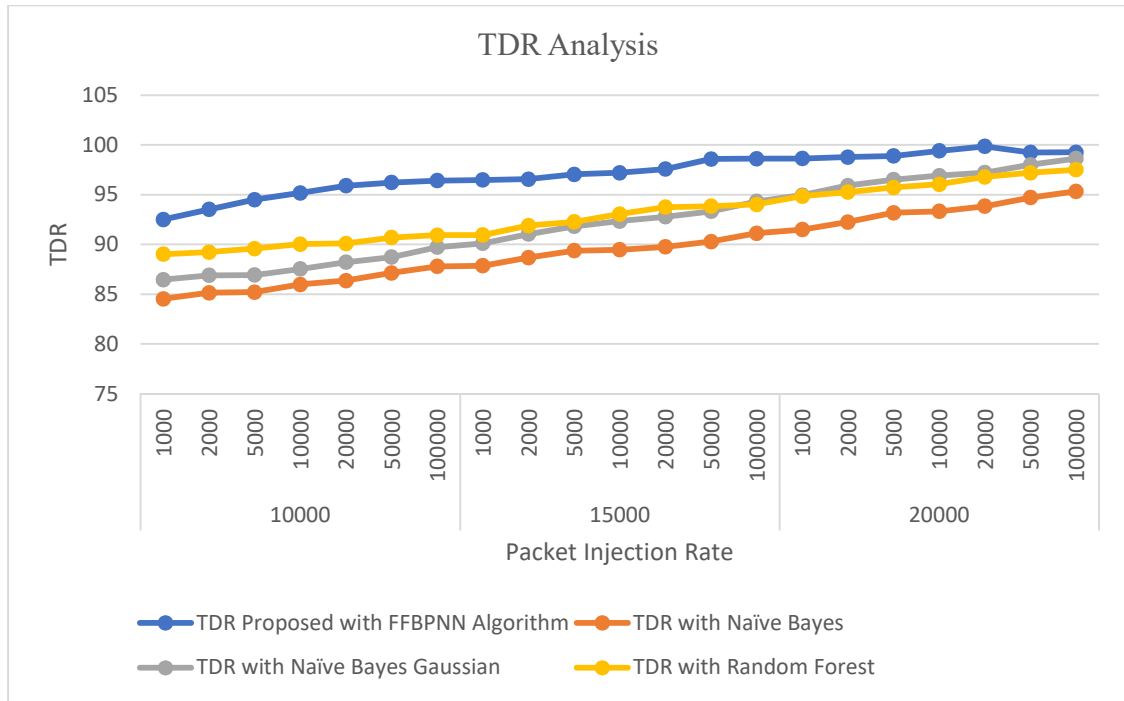


Figure 6.3 Evaluation of TDR Based on Different Classifiers

Figure 6.3 showcases the comparative analysis of the proposed technique using FFBPNN and the existing techniques. The graphical results show that the TDR using the proposed approach is better than the existing techniques. The results show that TDR using the proposed technique remains higher and exhibits better outcomes than the existing approaches.

Table 6.4 Evaluation of Jitter Based on Different Classifiers

| Packet Injection Rate (Packets/sec) | Total number of Simulations | Jitter Proposed with FFBPNN Algorithm | Jitter with Naïve Bayes | Jitter with Naïve Bayes Gaussian | Jitter with Random Forest |
|-------------------------------------|-----------------------------|---------------------------------------|-------------------------|----------------------------------|---------------------------|
| 10000                               | 1000                        | 272.32                                | 273.322                 | 281.554                          | 273.1177                  |
| 10000                               | 2000                        | 263.22                                | 272.21658               | 274.111                          | 272.008194                |
| 10000                               | 5000                        | 239.7988                              | 266.66472               | 268.668                          | 266.456334                |
| 10000                               | 10000                       | 217.3776                              | 261.11286               | 263.225                          | 260.904474                |
| 10000                               | 20000                       | 191.9564                              | 245.361                 | 247.782                          | 245.152614                |
| 10000                               | 50000                       | 169.5352                              | 229.60914               | 232.339                          | 229.400754                |
| 10000                               | 100000                      | 147.114                               | 213.85728               | 216.896                          | 213.648894                |

|       |        |          |           |         |            |
|-------|--------|----------|-----------|---------|------------|
| 15000 | 1000   | 123.6928 | 198.10542 | 201.453 | 197.897034 |
| 15000 | 2000   | 100.2716 | 182.35356 | 186.01  | 182.145174 |
| 15000 | 5000   | 74.8504  | 158.4417  | 162.567 | 158.233314 |
| 15000 | 10000  | 62.4292  | 142.68984 | 147.124 | 142.481454 |
| 15000 | 20000  | 48.008   | 126.93798 | 131.681 | 126.729594 |
| 15000 | 50000  | 45.5868  | 116.28612 | 121.238 | 116.077734 |
| 15000 | 100000 | 43.1656  | 97.47426  | 102.795 | 97.265874  |
| 20000 | 1000   | 39.7444  | 77.6424   | 83.3523 | 77.434014  |
| 20000 | 2000   | 38.3232  | 57.81054  | 63.9093 | 57.602154  |
| 20000 | 5000   | 31.902   | 37.97868  | 44.4663 | 37.770294  |
| 20000 | 10000  | 26.4808  | 28.34682  | 35.0233 | 28.138434  |
| 20000 | 20000  | 20.0596  | 20.75496  | 23.5803 | 20.546574  |
| 20000 | 50000  | 15.6384  | 17.2431   | 24.1373 | 21.114714  |
| 20000 | 100000 | 10.462   | 11.365    | 15.6943 | 12.2577    |

Table 6.4 shows the evaluation of jitter using the proposed approach and the existing classifiers. Initially, the proposed work exhibited a jitter of 26.44sec while a jitter using Naïve Bayes Gaussian is 31.36sec and using the Random Forest is 32.25sec. As the number of simulation rounds is increased along with the packet injection rate, a steep rise in the jitter is observed for all the techniques. This observation is further illustrated using Figure 6.4. The tabulated values also show that at a high packet injection rate of 20000 packets per sec, the jitter of the proposed technique remains lower than the existing ones. Thus, jitter analysis shows that using the proposed technique provides much more efficient V2V communication in comparison to the existing techniques.

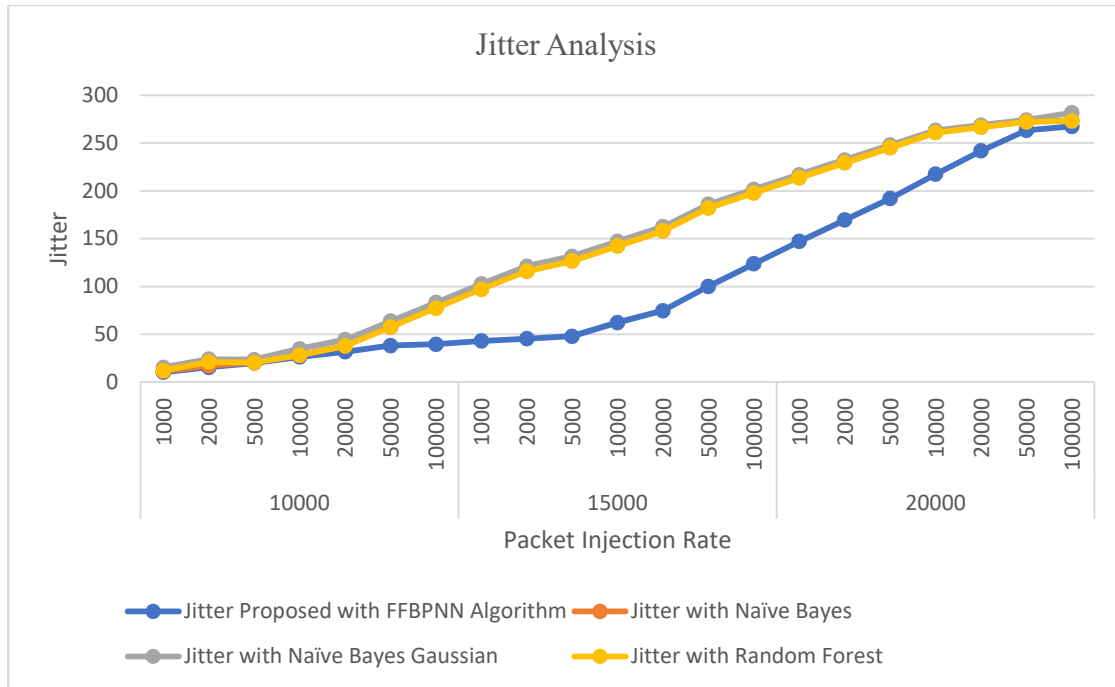


Figure 6.4 Evaluation of Jitter Based on Different Classifiers

Figure 6.4 shows the comparison and the graphical results showcase that the jitter using the proposed approach is better than the existing techniques. The evaluation results depict that jitter using the proposed technique falls down with an increase in simulation rounds but, outcomes are better than the existing approaches.

### 6.3 Comparative Analysis for QoS Measurement

To justify the success of the proposed work its performance is compared against two existing studies cited as Yuvraj et al. [47] and Kim et al. [55]. The framework presented by Yuvraj et al. was aimed at addressing the asymmetry in the network to extend the QoS and designed a congestion-aware routing. Kim et al. extended the OLSR and AODV protocols with the intelligent multi-criteria-based decision-making approach in a mobile node-based communication network. In contrast to these, the proposed work integrated machine learning, fuzzy logic, and several other criteria in the designed framework. Thus, the quality of all the techniques is evaluated and presented in the below tables and graphs for throughput, PDR, TDR, and jitter.

Table 6.5 shows the analysis of the proposed technique with other state-of-the-art techniques for the calculation of throughput. It is clearly visible that the average throughput value by Yuvraj et al. and Kim et al. is 18969.4 kbps and 18545.4 Kbps respectively. The proposed technique shows average throughput of about 19219.39 kbps.

Table 6.5 Comparative Analysis of Throughput

| Number of Nodes | Throughput Proposed | Throughput Yuvraj et al. | Throughput Kim et al. | Proposed over Yuvraj et al. | Proposed by Kim et al. |
|-----------------|---------------------|--------------------------|-----------------------|-----------------------------|------------------------|
| 50              | 19598.7             | 19329                    | 18632                 | 1.395313                    | 5.188386               |
| 60              | 19731.3             | 19573                    | 19041                 | 0.808767                    | 2.793971               |
| 70              | 19017.4             | 18570                    | 18333                 | 2.409262                    | 1.292751               |
| 80              | 18785.8             | 18286                    | 17832                 | 2.733239                    | 2.545985               |
| 90              | 18594.5             | 18525                    | 18093                 | 0.375169                    | 2.387664               |
| 100             | 18929.3             | 18666                    | 18468                 | 1.410586                    | 1.072125               |
| 200             | 19048.4             | 18746                    | 18449                 | 1.613144                    | 1.609843               |
| 300             | 19113.9             | 19036                    | 18652                 | 0.409225                    | 2.05876                |
| 400             | 19979.4             | 19651                    | 18982                 | 1.671162                    | 3.524392               |
| 500             | 19395.2             | 19312                    | 18972                 | 0.43082                     | 1.792115               |
| Average         | 19219.39            | 18969.4                  | 18545.4               | 1.32567                     | 2.4266                 |

Figure 6.5 shows the comparative analysis of QoS measurement. It is clearly visible that the average throughput value using the proposed technique is better than the existing techniques such as Yuvraj et al. and Kim et al. Thus, the proposed technique shows better results in comparison to existing techniques.

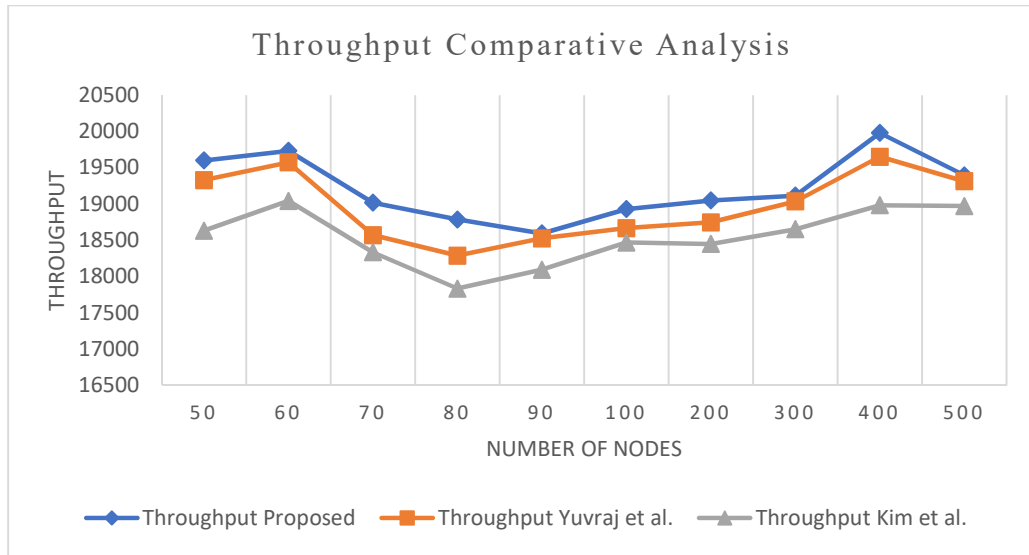


Figure 6.5 Comparative Analysis of Throughput

Figure 6.5 depicts the improvement analysis of the said technique in comparison to other techniques. The improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 1.32% and 2.42%. Thus, the proposed technique shows better results in comparison to other techniques. Thus, the proposed technique proved efficient and gave prominent results in comparison to other techniques.

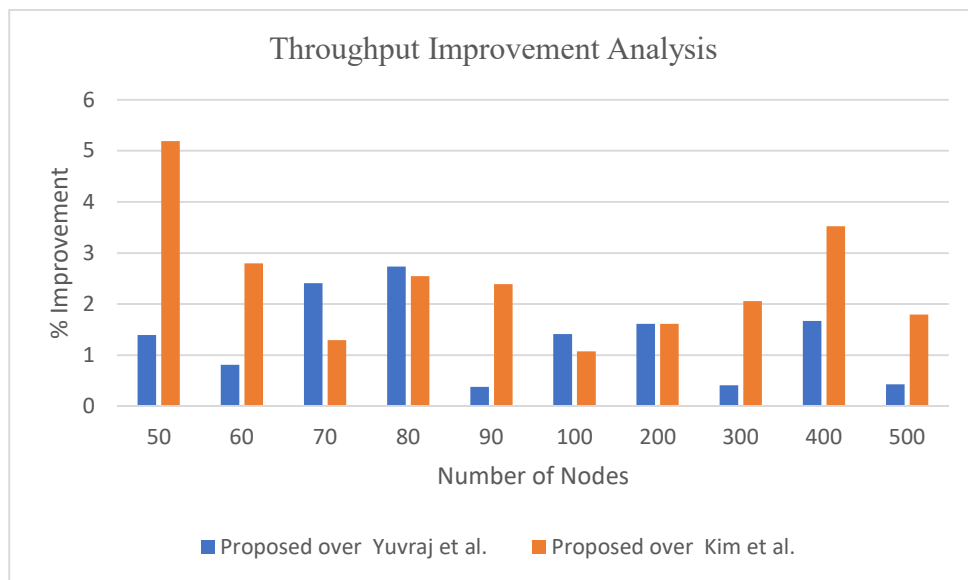


Figure 6.6 Improvement Analysis of Throughput

Table 6.6 Comparative Analysis of PDR

| Number of Nodes | PDR Proposed | PDR Yuvraj et al. | PDR Kim et al. | Proposed over Yuvraj et al. | Proposed by Kim et al. |
|-----------------|--------------|-------------------|----------------|-----------------------------|------------------------|
| 50              | 0.9335652    | 0.89456           | 0.8759         | 4.3602665                   | 6.583536933            |
| 60              | 0.9397668    | 0.88641           | 0.8715         | 6.019426676                 | 7.833253012            |
| 70              | 0.9412356    | 0.88721           | 0.8752         | 6.089381319                 | 7.545201097            |
| 80              | 0.9430716    | 0.89939           | 0.8848         | 4.856802944                 | 6.58584991             |
| 90              | 0.944112     | 0.89321           | 0.8846         | 5.698771845                 | 6.727560479            |
| 100             | 0.9484368    | 0.92224           | 0.8958         | 2.84056211                  | 5.875954454            |
| 200             | 0.947121     | 0.91437           | 0.8865         | 3.581810427                 | 6.838240271            |
| 300             | 0.9472536    | 0.91114           | 0.8848         | 3.963562131                 | 7.058499096            |
| 400             | 0.9452646    | 0.91658           | 0.8904         | 3.129524973                 | 6.161792453            |
| 500             | 0.9395016    | 0.90409           | 0.8958         | 3.91682244                  | 4.878499665            |
| Average         | 0.9429329    | 0.90292           | 0.88453        | 4.44569314                  | 6.60883874             |

Table 6.6 shows the analysis of the said technique with the other existing techniques for PDR. It is clearly seen that the PDR value by Yuvraj et al. and Kim et al. is 0.90 and 0.88 respectively. The proposed technique shows an average PDR of about 0.94. Thus, the proposed mechanism showcases improved results in comparison to other existing techniques.



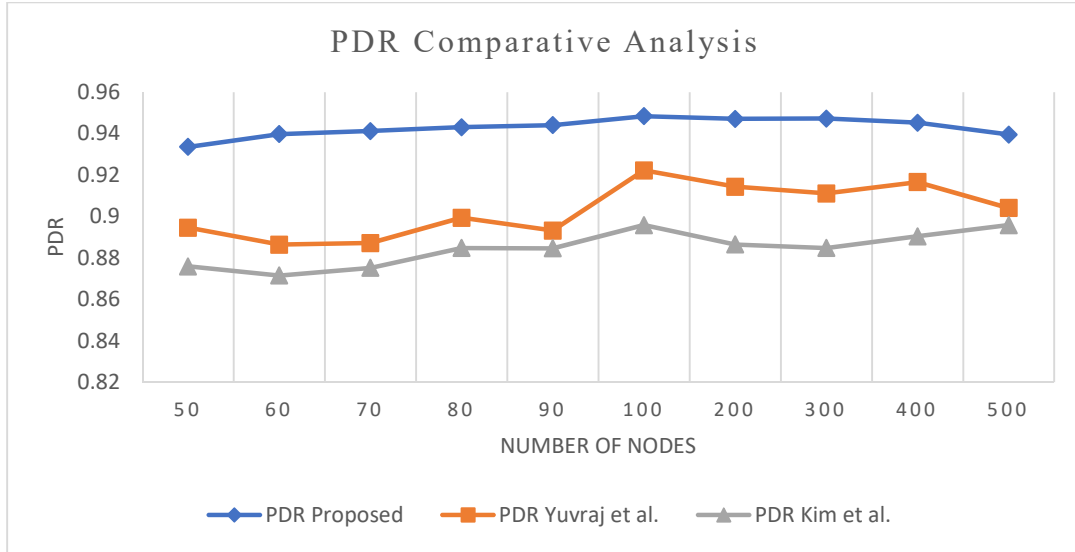


Figure 6.7 Comparative Analysis of PDR

Figure 6.7 shows the analysis of the said technique with the other existing techniques with existing techniques for PDR measurement. The PDR using Yuvraj et al. follows the zigzag pattern and PDR is less than the proposed technique while results using the Kim et al. are also lower than the proposed approach. Thus, the proposed mechanism showcases improved results in comparison to other existing techniques.

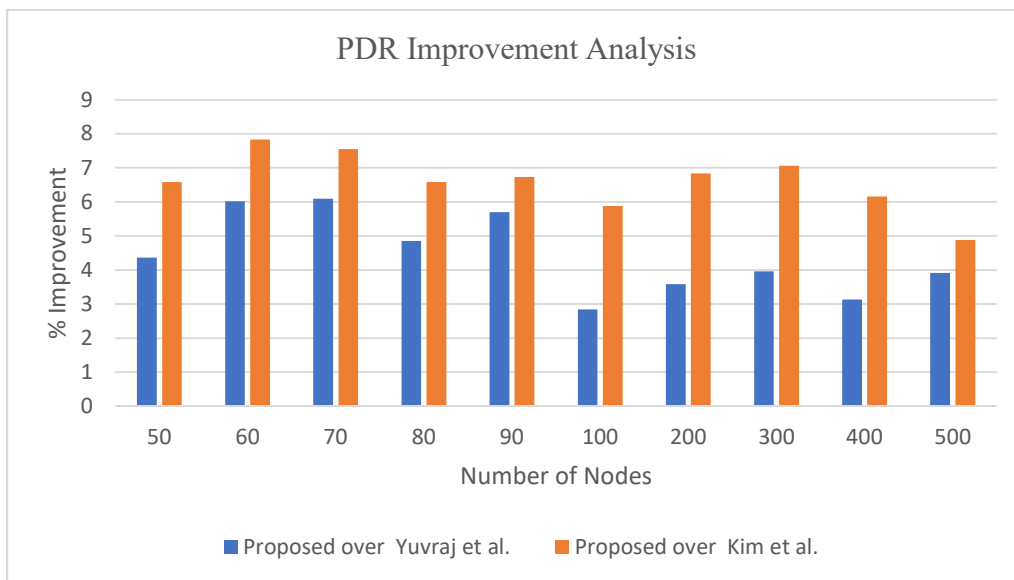


Figure 6.8 Improvement Analysis of PDR

It is clearly visible that the average PDR value using the said technique is better than the existing techniques such as Yuvraj et al. and Kim et al. The improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 4.44% and 6.6%. The improvement analysis shows the robustness of the proposed technique.

Table 6.7 Comparative Analysis of TDR

| Number of Nodes | TDR Proposed | TDR Yuvraj et al. | TDR Kim et al. | TDR over Yuvraj et al. | TDR over Kim et al. |
|-----------------|--------------|-------------------|----------------|------------------------|---------------------|
| 50              | 90.0711618   | 85.0836           | 87.3169        | 5.861944               | 3.154293            |
| 60              | 91.4274465   | 85.9344           | 87.6662        | 6.392085               | 4.290421            |
| 70              | 91.321721    | 86.7938           | 88.5429        | 5.216885               | 3.138436            |
| 80              | 94.4055816   | 87.6617           | 89.4283        | 7.693044               | 5.565681            |
| 90              | 95.5938542   | 88.5383           | 90.3226        | 7.968876               | 5.836064            |
| 100             | 94.5197927   | 89.4237           | 91.2258        | 5.698784               | 3.610816            |
| 200             | 96.73555     | 90.318            | 92.1381        | 7.105547               | 4.989789            |
| 300             | 96.8723766   | 91.2211           | 93.0594        | 6.195091               | 4.097318            |
| 400             | 97.1187175   | 92.1334           | 93.99          | 5.411028               | 3.328744            |
| 500             | 98.47838     | 93.0547           | 94.9299        | 7.977771               | 5.844784            |
| Average         | 94.6544581   | 89.0163           | 90.862         | 6.55211                | 4.38563             |

Table 6.7 depicts the analysis of the said technique with the other existing techniques for TDR. It is clearly seen that the average TDR value by Yuvraj et al. and Kim et al. is 89.016% and 90.86%, respectively. The proposed technique shows an average TDR of about 94.65%. The improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 6.5% and 4.3%. Thus, the proposed mechanism showcases improved results in comparison to other existing techniques.

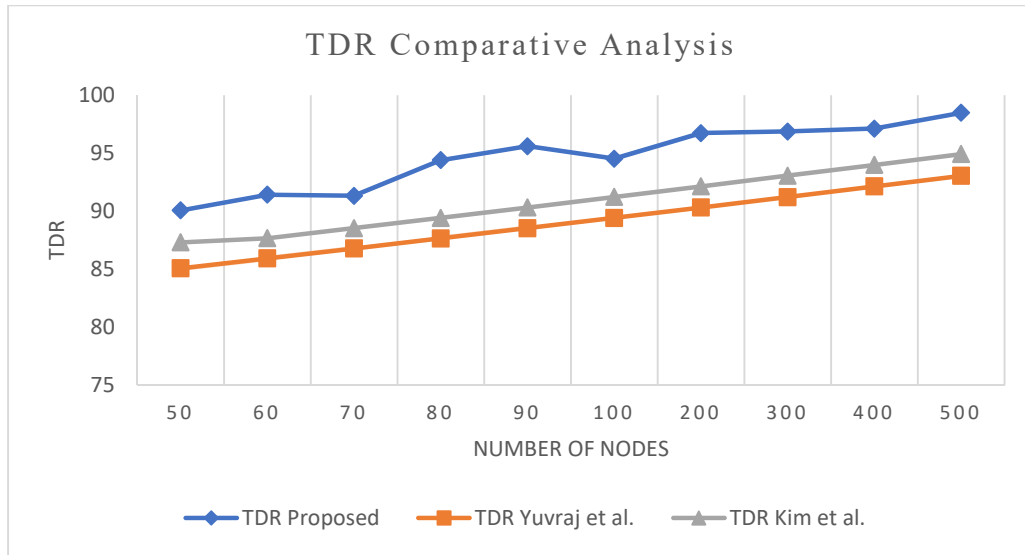


Figure 6.9 Comparative Analysis of TDR

Figure 6.9 depicts the comparative analysis of the proposed mechanism with the other existing techniques for TDR measurement. The TDR using Yuvraj et al. is less than the proposed technique while results using Kim et al. are also lower than the proposed approach as well as Yuvraj et al. Thus, the proposed mechanism showcases improved results in comparison to other existing techniques.

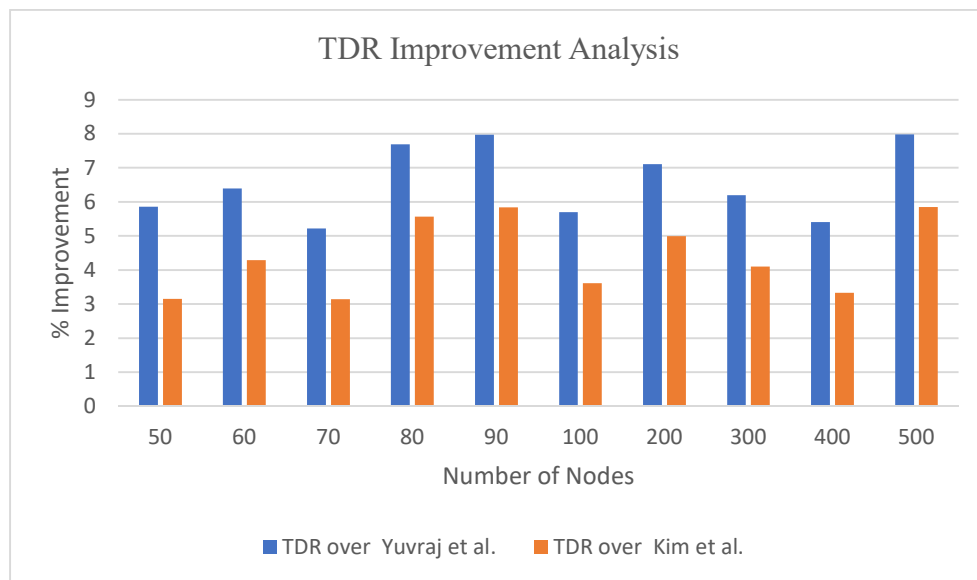


Figure 6.10 Improvement Analysis of TDR

It is clearly visible that the average TDR value using the said technique is better than the existing techniques like Yuvraj et al. and Kim et al. The improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 6.5% and 4.3%. The improvement analysis shows the robustness of the proposed technique.

Table 6.8 Comparative Analysis of Jitter

| Number of Nodes | Jitter Proposed | Jitter Yuvraj et al. | Jitter Kim et al. | Jitter over Yuvraj et al. | Jitter over Kim et al. |
|-----------------|-----------------|----------------------|-------------------|---------------------------|------------------------|
| 50              | 10.2911618      | 15.3664              | 11.4462           | 33.02815363               | 10.09101885            |
| 60              | 13.5111618      | 15.9104              | 14.9902           | 15.07968499               | 9.866700911            |
| 70              | 16.8311618      | 18.4544              | 17.5342           | 8.795941347               | 4.009525385            |
| 80              | 17.1511618      | 19.9984              | 20.0782           | 14.23732999               | 14.57819028            |
| 90              | 19.4711618      | 23.5424              | 22.6222           | 17.29321649               | 13.92896447            |
| 100             | 25.7911618      | 30.0864              | 30.1662           | 14.27634479               | 14.50311342            |
| 200             | 30.1111618      | 38.6304              | 34.7102           | 22.05319696               | 13.24981763            |
| 300             | 36.4311618      | 45.1744              | 41.2542           | 19.35440913               | 11.69102346            |
| 400             | 40.7511618      | 51.7184              | 48.7982           | 21.2056796                | 16.49044063            |
| 500             | 54.0711618      | 56.2624              | 56.3422           | 3.894676018               | 4.030794325            |
| Average         | 26.4411618      | 31.5144              | 29.7942           | 16.9218633                | 11.24395894            |

Table 6.8 shows the comparative analysis of the proposed technique with the existing techniques for Jitter. It is clearly seen that the average Jitter value for 500 nodes by Yuvraj et al. and Kim et al. is 31.51sec and 29.79sec, respectively. In comparison to these, the proposed technique shows an average jitter of about 26.44sec.

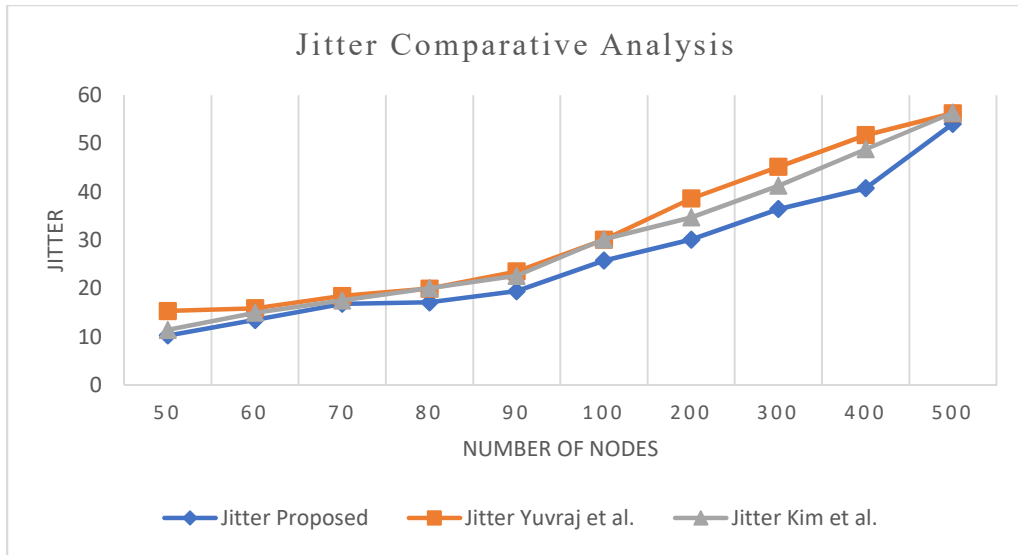


Figure 6.11 Comparative Analysis of Jitter

Figure 6.11 depicts the analysis of the said technique with the other existing techniques for Jitter measurement. The Jitter using the Yuvraj et al. is higher than the proposed technique while results using the Kim et al. are also higher than the proposed approach as well as Yuvraj et al. The higher value of the jitter indicates high noise and more distortion. Thus, the proposed mechanism showcases improved results in comparison to other existing techniques.

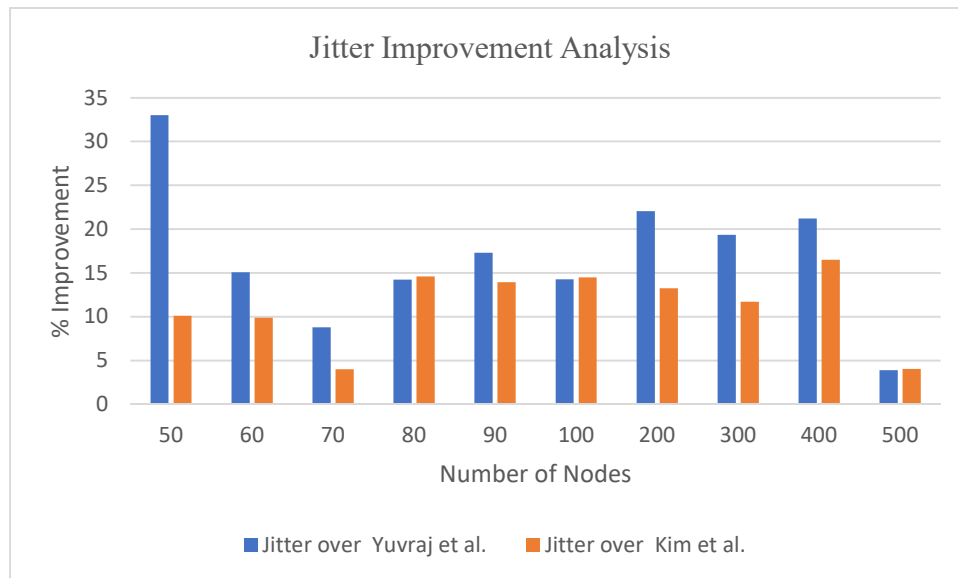


Figure 6.12 Improvement Analysis of Jitter

The average jitter improvement of the proposed approach to the Yuvraj et al. and Kim et al. is 16.92% and 11.24%. Thus, the proposed technique shows better results in comparison to existing techniques. The improvement analysis shows the robustness of the proposed technique.

#### 6.4 Summary of the Chapter

This chapter elaborates on the results and discussion using the different performance metrics. The QoS measurement also explained the effectiveness of the implemented technique in VANET. In other words, it also establishes that better QoS measurements also reflect the involvement of trusted nodes during root discovery. Because in case there would be compromised nodes in the route, there may be instances of network failure that would have degraded the QoS. Hence, the nodes are labelled as trusted nodes when QoS is higher than the existing works.

## Chapter 7: Conclusion

---

The chapter is dedicated to presenting the drawn conclusions to the detailed research summarized in the earlier chapters of the thesis. The research study aims are designing an improved and secure V2V communication based on the strengths of machine learning techniques. In the process, the work involves labeling of the nodes which is based on the QoS parameters. Thus, trust is established in V2V communication based on the performance of the node in all the contributed routes. This is analyzed in terms of throughput, PDR, TDR, and delay.

### 7.1 Conclusion

The vehicles transfer the information through the communication channel. The evaluation of the channel can only be done once the data is transferred from the source vehicle to the terminal vehicle. The proposed research work has utilized MATLAB as the simulation tool for the implementation of the designed algorithmic architectures in VANET. The performance metrics such as TPR, FPR, and accuracy have been determined for different training to the testing set distributions such as 70-30, 80-20, and 90-10.

- For 70:30 dataset distribution, TPR and FPR results are contrasted with the other state-of-the-art techniques such as Naïve Bayes and Random Forest. The average value for TPR and FPR using the proposed technique is 0.96 and 0.065 respectively while the existing technique shows a TPR of about 0.93 for both Naïve Bayes and 0.94 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.084 and 0.08 respectively. Thus, the proposed technique shows prominent FPR and TPR in comparison to Naïve Bayes and Random Forest. Additionally, the confusion matrix has been also realized, and it is observed that out of the total 58550 records were representing DDoS attacks with high-class accuracy for DDoS shown by 54000 records.

- For 80:20 distribution ratios, the average value of TPR and FPR using the proposed technique is about 0.96 and 0.065 respectively while the existing technique shows TPR of about 0.93 for both Naïve Bayes and 0.94 for Random Forest. Consequently, FPR using the existing technique such as Naïve Bayes and Random Forest is 0.09 and 0.07 respectively. Thus, the proposed technique shows prominent FPR and TPR for 80:20 ratios in comparison to Naïve Bayes and Random Forest. The confusion matrix for the 80:20 ratio shows that out of the total records, 59150 records were representing DDoS attacks with high-class accuracy while others show accuracy for 6300 records.
- For 90:10 distribution ratios, the average value of TPR and FPR using the proposed technique is about 0.97 and 0.053 respectively while the existing technique shows TPR of about 0.950 for both Naïve Bayes and 0.958 for Random Forest. Consequently, FPR using the existing technologies such as Naïve Bayes and Random Forest is 0.079 and 0.071 respectively. Thus, the proposed technique shows prominent FPR and TPR using a 90:10 ratio in comparison to Naïve Bayes and Random Forest. The confusion matrix for the 90:10 ratio shows that out of the total records, 59700 records were representing DDoS attacks with high-class accuracy while others show accuracy for 6300 records.

Further, the efficiency of the proposed technique has been showcased by comparing the QoS observed against the existing techniques. The number of vehicle nodes used in the analysis is restricted from 50 to 500 and the throughput, PDR, TDR, and jitter analysis is performed.

- In the case of throughput analysis, the average throughput using the proposed approach is 19219.39 Kbps, using Yuvraj et al. approach is 1896.4 Kbps, and using Kim et al. approach is 18545.4 Kbps. This shows that the throughput using the proposed technique is better than the existing techniques. In comparison to Yuvraj et al. and Kim et al., the proposed approach exhibited an average improvement of 1.3% and 2.4%, respectively. Thus, the proposed technique showcases improved results in comparison to other existing techniques.



- PDR analysis using the proposed approach and the existing techniques has been presented. It is seen that the average PDR using the proposed approach is 0.9429 while Yuvraj et al. is 0.9029 and Kim et al. is 0.88453 showing that the PDR using the proposed technique is better than the existing techniques. Thus, an average improvement of 4.44% and 6.6% is shown by the proposed approach over Yuvraj et al. and Kim et al. Thus, the proposed technique showcases improved results in comparison to other existing techniques.
- The average TDR observed using the proposed work is 94.65% while the TDR exhibited by Yuvraj et al. and Kim et al. is 89.016% and 90.86%, respectively. Thus, the TDR improvement of the proposed approach over Yuvraj et al. is 6.55%, and Kim et al. is 4.38%.
- Similarly, the average Jitter value exhibited by the proposed is 26.44sec, Yuvraj et al. is 31.51sec and Kim et al. are 29.39sec. Thus, exhibits an overall improvement of 16.92% over Yuvraj et al. and 11.24% over Kim et al. The improvement analysis shows the robustness of the proposed technique.

The overall simulation study concluded that the higher QoS measurements have been demonstrated by the proposed work in comparison to the two existing studies of Yuvraj et al. and Kim et al. The higher QoS measurements illustrated by the proposed work also reflect the involvement of trusted nodes in the route and thus justify the success of the proposed work. In other words, the nodes involved in the route are labelled as trusted nodes only when they could deliver an acceptable quality of service.

## 7.2 Future Scope

The presented work is aimed to distinguish trusted and un-trusted nodes so as to improve the reliability aspect in VANET architecture. In the process, a packet-efficient prevention architecture was developed to enhance secure V2V communication based on Machine Learning. In the evaluation of the present work, various ML algorithms are evaluated for the two most common attacks observed in VANETs. In the future, the present architecture can be extended to provide defence against more attacks in addition to DDoS

and Replay attacks. In addition to this, the author will look forward to some other insect or animal swarm algorithms that can be integrated to refine the present work or deep learning concepts can be involved with the neural network architecture to improve the performance of the present work.

## REFERENCES

---

- [1] Khalaf, O. I., & Sabbar, B. M. (2019). An overview on wireless sensor networks and finding optimal location of nodes. *Periodicals of Engineering and Natural Sciences*, 7(3), 1096-1101.
- [2] Obeidat, H., Shuaieb, W., Obeidat, O., & Abd-Alhameed, R. (2021). A review of indoor localization techniques and wireless technologies. *Wireless Personal Communications*, 119(1), 289-327.
- [3] Elhoseny, M., & Hassanien, A. E. (2019). Secure data transmission in WSN: an overview. *Dynamic wireless sensor networks*, 115-143.
- [4] Ranjan, P., & Ahirwar, K. K. (2011, January). Comparative study of vanet and manet routing protocols. In Proceedings of the International Conference on Advanced computing and communication Technologies (ACCT 2011).
- [5] Hartenstein, H., & Laberteaux, K. P. (2008). A tutorial survey on vehicular ad hoc networks. *Communications Magazine, IEEE*, 46(6), 164-171.
- [6] Bako, B., & Weber, M. (2011). Efficient information dissemination in VANETs. INTECH Open Access Publisher
- [7] Kumar, R., & Dave, M. (2012). A review of various vanet data dissemination protocols. *International Journal of u-and eService, Science and Technology*, 5(3), 27-44.
- [8] Kolte, S. R., & Madankar, M. S. (2014, April). Adaptive congestion control for transmission of safety messages in VANET. In *International Conference for Convergence for Technology-2014* (pp. 1-5). IEEE.
- [9] Ducourthial, B., Khaled, Y., & Shawky, M. (2007). Conditional transmissions: Performance study of a new communication strategy in VANET. *IEEE Transactions on Vehicular Technology*, 56(6), 3348-3357.
- [10] Maier, M. W., Emery, D., & Hilliard, R. (2001). Software architecture: Introducing IEEE standard 1471. *Computer*, 34(4), 107-109.

- [11] Hilliard, R. (2000). Ieee-std-1471-2000 recommended practice for architectural description of software-intensive systems. *IEEE*, <http://standards.ieee.org>, 12(16-20), 2000.
- [12] Emery, D., & Hilliard, R. (2009, September). Every architecture description needs a framework: Expressing architecture frameworks using ISO/IEC 42010. In *2009 Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture* (pp. 31-40). IEEE.
- [13] Schroth, C., Kosch, T., Strassberger, M., & Bechler, M. (2012). *Automotive internetworking*. John Wiley & Sons.
- [14] Moustafa, H., Zhang, Y. (2009)  *Vehicular Networks: Techniques, Standards, and Applications*, Boca Raton, Fla, USA CRC Press.
- [15] CAR 2 CAR Communication Consortium Manifesto, 2007, [http://elib.dlr.de/48380/1/C2C-CC\\_manifesto\\_v1.1.pdf](http://elib.dlr.de/48380/1/C2C-CC_manifesto_v1.1.pdf)
- [16] Raw, R. S., Kumar, M., & Singh, N. (2013). Security challenges, issues and their solutions for VANET. *International journal of network security & its applications*, 5(5), 95.
- [17] Gupta, N., Jain, A., Vaisla, K. S., Kumar, A., & Kumar, R. (2021). Performance analysis of DSDV and OLSR wireless sensor network routing protocols using FPGA hardware and machine learning. *Multimedia Tools and Applications*, 80(14), 22301-22319.
- [18] Jain, R., & Kashyap, I. (2019). An QoS aware link defined OLSR (LD-OLSR) routing protocol for MANETs. *Wireless Personal Communications*, 108(3), 1745-1758.
- [19] Zhang, J., & Sun, Z. (2016, June). Assessing multi-hop performance of reactive routing protocols in wireless sensor networks. In *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)* (pp. 444-449). IEEE.
- [20] Sallam, G., & Mahmoud, A. (2015, April). Performance evaluation of OLSR and AODV in VANET cloud computing using fading model with SUMO and NS3. In *2015 International Conference on Cloud Computing (ICCC)* (pp. 1-5). IEEE.

- [21] Spaho, E., Ikeda, M., Barolli, L., & Xhafa, F. (2013). Performance comparison of OLSR and AODV protocols in a VANET crossroad scenario. In *Information Technology Convergence* (pp. 37-45). Springer, Dordrecht.
- [22] Poonia, R., Sanghi, A. K., & Singh, D. (2011). DSR routing protocol in wireless ad-hoc networks: Drop analysis. *International Journal of Computer Applications*, 14(7), 18-21.
- [23] Darabkh, K. A., Alfawares, M. G., & Althunibat, S. (2019). MDRMA: Multi-data rate mobility-aware AODV-based protocol for flying ad-hoc networks. *Vehicular Communications*, 18, 100163.
- [24] Jain, K., & Jeyakumar, A. (2016, April). An RSU Based Approach: A solution to overcome major issues of Routing in VANET. In *2016 International Conference on Communication and Signal Processing (ICCSP)* (pp. 1265-1269). IEEE.
- [25] Gasmi, R., Aliouat, M., & Seba, H. (2020). A stable link based zone routing protocol (SL-ZRP) for internet of vehicles environment. *Wireless Personal Communications*, 112(2), 1045-1060.
- [26] Eze, E. C., Zhang, S. J., Liu, E. J., & Eze, J. C. (2016). Advances in vehicular ad-hoc networks (VANETs): Challenges and road-map for future development. *International Journal of Automation and Computing*, 13(1), 1-18.
- [27] Omar, H. A., Lu, N., & Zhuang, W. (2016). Wireless access technologies for vehicular network safety applications. *IEEE Network*, 30(4), 22-26.
- [28] Sasongko, A. T., Jati, G., Hardian, B., & Jatmiko, W. (2020). The Reliability of Routing Protocols as an Important Factor for Road Safety Applications in VANET-based Autonomous Cars. *Journal of Computer Science*, 16(6), 768-783.
- [29] Khatri, S., Vachhani, H., Shah, S., Bhatia, J., Chaturvedi, M., Tanwar, S., & Kumar, N. (2021). Machine learning models and techniques for VANET based traffic management: Implementation issues and challenges. *Peer-to-Peer Networking and Applications*, 14(3), 1778-1805.
- [30] Sheikh, M. S., & Liang, J. (2019). A comprehensive survey on VANET security services in traffic management system. *Wireless Communications and Mobile Computing*, 2019.

- [31] Hamdi, M. M., Audah, L., Rashid, S. A., Mohammed, A. H., Alani, S., & Mustafa, A. S. (2020, June). A review of applications, characteristics and challenges in vehicular ad hoc networks (VANETs). In *2020 international congress on human-computer interaction, optimization and robotic applications (HORA)* (pp. 1-7). IEEE.
- [32] Ramli, N. I., Shawal, S., Ibrahim, M. F., & Rawi, M. I. M. (2021, November). Network Performance Analysis of Content Delivery for V2R Communication in VANET Urban Mobility. In *2021 IEEE International Conference on Computing (ICOCO)* (pp. 129-133). IEEE.
- [33] Manipriya, S., Mala, C., & Mathew, S. (2020). A collaborative framework for traffic information in vehicular adhoc network applications. *Journal of Internet Services and Information Security (JISIS)*, 10(3), 93-109.
- [34] Kumar, R. R., Indumathi, P., & Satheeswaran, C. (2018, March). Spectrum and Traffic Aware Routing Protocol with cooperative information collection method for CR-VANET. In *2018 International Conference on Recent Trends in Electrical, Control and Communication (RTECC)* (pp. 35-40). IEEE.
- [35] Al Junaid, M. A. H., Syed, A. A., Warip, M. N. M., Azir, K. N. F. K., & Romli, N. H. (2018). Classification of security attacks in VANET: A review of requirements and perspectives. In *MATEC web of conferences* (Vol. 150, p. 06038). EDP Sciences.
- [36] Abassi, R. (2019). VANET security and forensics: Challenges and opportunities. *Wiley Interdisciplinary Reviews: Forensic Science*, 1(2), e1324.
- [37] Sheikh, M. S., & Liang, J. (2019). A comprehensive survey on VANET security services in traffic management system. *Wireless Communications and Mobile Computing*, 2019
- [38] Kulkarni, S. A., & Rao, G. R. (2010, August). Vehicular ad hoc network mobility models applied for reinforcement learning routing algorithm. In *International Conference on Contemporary Computing* (pp. 230-240). Springer, Berlin, Heidelberg.

- [39] Desjardins, C., & Chaib-Draa, B. (2011). Cooperative adaptive cruise control: A reinforcement learning approach. *IEEE Transactions on intelligent transportation systems*, 12(4), 1248-1260.
- [40] Dogru, N., & Subasi, A. (2012, May). Traffic accident detection by using machine learning methods. In *Third International Symposium on Sustainable Development (ISSD'12)* (p. 467).
- [41] Bento, L. C., Parafita, R., Santos, S., & Nunes, U. (2013, October). Intelligent traffic management at intersections: Legacy mode for vehicles not equipped with V2V and V2I communications. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)* (pp. 726-731). IEEE.
- [42] Lai, W. K., Lin, M. T., & Yang, Y. H. (2015). A machine learning system for routing decision-making in urban vehicular ad hoc networks. *International Journal of Distributed Sensor Networks*, 11(3), 374391.
- [43] Perfecto, C., Del Ser, J., & Bennis, M. (2017). Millimeter-wave V2V communications: Distributed association and beam alignment. *IEEE Journal on Selected Areas in Communications*, 35(9), 2148-2162.
- [44] Peng, H., Liang, L., Shen, X., & Li, G. Y. (2018). Vehicular communications: A network layer perspective. *IEEE Transactions on Vehicular Technology*, 68(2), 1064-1078.
- [45] Sim, G. H., Klos, S., Asadi, A., Klein, A., & Hollick, M. (2018). An online context-aware machine learning algorithm for 5G mmWave vehicular communications. *IEEE/ACM Transactions on Networking*, 26(6), 2487-2500.
- [46] Uchida, N., Hashimoto, R., Sato, G., & Shibata, Y. (2018, September). Adaptive Array Antenna Controls with Machine Learning Based Image Recognition for Vehicle to Vehicle Networks. In *International Conference on Network-Based Information Systems* (pp. 521-530). Springer, Cham.
- [47] Yuvaraj, N., & Thangaraj, P. (2018). An extended cross layer approach handling asymmetry with quality of service, congestion avoidance and congestion aware routing. *Journal of Computational and Theoretical Nanoscience*, 15(2), 676-684.

- [48] Atallah, R. F., Assi, C. M., & Khabbaz, M. J. (2018). Scheduling the operation of a connected vehicular network using deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 20(5), 1669-1682.
- [49] Kumar, S., Singh, K., Kumar, S., Kaiwartya, O., Cao, Y., & Zhou, H. (2019). Delimitated anti jammer scheme for Internet of vehicle: Machine learning based security approach. *IEEE Access*, 7, 113311-113323.
- [50] Wu, W., Li, R., Xie, G., An, J., Bai, Y., Zhou, J., & Li, K. (2019). A survey of intrusion detection for in-vehicle networks. *IEEE Transactions on Intelligent Transportation Systems*, 21(3), 919-933.
- [51] Yee, R., Chan, E., Cheng, B., & Bansal, G. (2018, June). Collaborative perception for automated vehicles leveraging vehicle-to-vehicle communications. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1099-1106). IEEE.
- [52] Souhir, F. E. K. I., Belghith, A., & Zarai, F. (2019, June). A reinforcement learning-based radio resource management algorithm for D2D-based V2V communication. In *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)* (pp. 1367-1372). IEEE.
- [53] Huang, C., Molisch, A. F., He, R., Wang, R., Tang, P., & Zhong, Z. (2019). Machine-learning-based data processing techniques for vehicle-to-vehicle channel modeling. *IEEE Communications Magazine*, 57(11), 109-115.
- [54] Gyawali, S., & Qian, Y. (2019, May). Misbehavior detection using machine learning in vehicular communication networks. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)* (pp. 1-6). IEEE.
- [55] Kim, B. S., Roh, B., Ham, J. H., & Kim, K. I. (2020). Extended OLSR and AODV based on multi-criteria decision making method. *Telecommunication Systems*, 73, 241-257.
- [56] Patel, S. R., & Ajmeri, M. (2020). Machine learning approach for transportation services using vehicular ad hoc network. *Int J Trend Innov Res (IJTIIR)*, 2(2).
- [57] Ullah, Z., Al-Turjman, F., Mostarda, L., & Gagliardi, R. (2020). Applications of artificial intelligence and machine learning in smart cities. *Computer Communications*, 154, 313-323.



- [58] Turan, B., Uyrus, A., Koc, O. N., Kar, E., & Coleri, S. (2021, December). Machine Learning Aided Path Loss Estimator and Jammer Detector for Heterogeneous Vehicular Networks. In *2021 IEEE Global Communications Conference (GLOBECOM)* (pp. 1-6). IEEE.
- [59] Kang Kim, H., Becerra, R., Bolufé, S., Azurdia-Meza, C. A., Montejo-Sánchez, S., & Zabala-Blanco, D. (2021). Neuroevolution-based adaptive antenna array beamforming scheme to improve the V2V communication performance at intersections. *Sensors*, *21*(9), 2956.
- [60] Wong, S., Jiang, L., Walters, R., Molnár, T. G., Orosz, G., & Yu, R. (2021, May). Traffic forecasting using vehicle-to-vehicle communication. In *Learning for Dynamics and Control* (pp. 917-929). PMLR.
- [61] Pressas, A., Sheng, Z., Ali, F., Tian, D., & Nekovee, M. (2017, November). Contention-based learning MAC protocol for broadcast vehicle-to-vehicle communication. In *2017 IEEE Vehicular Networking Conference (VNC)* (pp. 263-270). IEEE.
- [62] Bahramnejad, S., & Movahhedinia, N. (2022). A reliability estimation framework for cognitive radio V2V communications and an ANN-based model for automating estimations. *Computing*, 1-25.
- [63] Zheng, K., Han, L., & Li, Y. (2022). Resource Allocation for Full-Duplex V2V Communications Based on Deep Reinforcement Learning. In *Artificial Intelligence in China* (pp. 586-592). Springer, Singapore.
- [64] Liu, N., Liu, M., Cao, J., Chen, G., & Lou, W. (2010, June). When transportation meets communication: V2P over VANETs. In *2010 IEEE 30th International Conference on Distributed Computing Systems* (pp. 567-576). IEEE.
- [65] Petit, J., Feiri, M., & Kargl, F. (2011, November). Spoofed data detection in VANETs using dynamic thresholds. In *2011 IEEE Vehicular Networking Conference (VNC)* (pp. 25-32). IEEE.
- [66] Villagrà, J., Milanés, V., Pérez, J., Godoy, J., Onieva, E., Alonso, J., ... & Garcia, R. (2011, February). A reinforcement learning modular control architecture for fully automated vehicles. In *International Conference on Computer Aided Systems Theory* (pp. 390-397). Springer, Berlin, Heidelberg.

- [67] Mecklenbräuker, C., Bernadó, L., Klemp, O., Kwoczek, A., Paier, A., Schack, M., ... & Zemen, T. (2012). Vehicle-to-Vehicle Communications. In *Pervasive Mobile and Ambient Wireless Communications* (pp. 577-608). Springer, London.
- [68] Kihei, B., Copeland, J. A., & Chang, Y. (2017, July). Automotive Doppler sensing: The Doppler profile with machine learning in vehicle-to-vehicle networks for road safety. In *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)* (pp. 1-5). IEEE.
- [69] Ye, H., Liang, L., Li, G. Y., Kim, J., Lu, L., & Wu, M. (2017). Machine learning for vehicular networks. *arXiv preprint arXiv:1712.07143*.
- [70] Ezianya, E., Tepe, K., Balador, A., Nwizege, K. S., & Jaimes, L. M. (2018, December). Malicious node detection in vehicular ad-hoc network using machine learning and deep learning. In *2018 IEEE Globecom Workshops (GC Wkshps)* (pp. 1-6). IEEE.
- [71] Liang, L., Ye, H., & Li, G. Y. (2018). Toward intelligent vehicular networks: A machine learning framework. *IEEE Internet of Things Journal*, 6(1), 124-135.
- [72] Ye, H., & Li, G. Y. (2018, May). Deep reinforcement learning for resource allocation in V2V communications. In *2018 IEEE International Conference on Communications (ICC)* (pp. 1-6). IEEE.
- [73] Tang, F., Kawamoto, Y., Kato, N., & Liu, J. (2019). Future intelligent and secure vehicular network toward 6G: Machine-learning approaches. *Proceedings of the IEEE*, 108(2), 292-307.
- [74] Saputra, Y. M., Hoang, D. T., Nguyen, D. N., Dutkiewicz, E., Mueck, M. D., & Srikanteswara, S. (2019, December). Energy demand prediction with federated learning for electric vehicle
- [75] Ye, H., Li, G. Y., & Juang, B. H. F. (2019). Deep reinforcement learning based resource allocation for V2V communications. *IEEE Transactions on Vehicular Technology*, 68(4), 3163-3173.
- [76] Aldossari, S. M., & Chen, K. C. (2019). Machine learning for wireless communication channel modeling: An overview. *Wireless Personal Communications*, 106(1), 41-70.

- [77] Alrehan, A. M., &Alhaidari, F. A. (2019, May). Machine learning techniques to detect DDoS attacks on VANET system: a survey. In *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)* (pp. 1-6). IEEE.
- [78] Ftaimi, S., &Mazri, T. (2020, March). A comparative study of Machine learning algorithms for VANET networks. In *Proceedings of the 3rd International Conference on Networking, Information Systems & Security* (pp. 1-8).
- [79] Sharma, P., & Liu, H. (2020). A machine-learning-based data-centric misbehavior detection model for the internet of vehicles. *IEEE Internet of Things Journal*, 8(6), 4991-4999.
- [80] Zhao, D., Qin, H., Song, B., Zhang, Y., Du, X., &Guizani, M. (2020). A reinforcement learning method for joint mode selection and power adaptation in the V2V communication network in 5G. *IEEE Transactions on Cognitive Communications and Networking*, 6(2), 452-463.
- [81] Tayyaba, S. K., Khattak, H. A., Almogren, A., Shah, M. A., Din, I. U., Alkhalifa, I., &Guizani, M. (2020). 5G vehicular network resource management for improving radio access through machine learning. *IEEE Access*, 8, 6792-6800.
- [82] Yang, Y., Gao, Z., Ma, Y., Cao, B., & He, D. (2020). Machine learning enabling analog beam selection for concurrent transmissions in millimeter-wave V2V communications. *IEEE Transactions on Vehicular Technology*, 69(8), 9185-9189.
- [83] Choi, D., Yim, J., Baek, M., & Lee, S. (2021). Machine learning-based vehicle trajectory prediction using v2v communications and on-board sensors. *Electronics*, 10(4), 420.
- [84] Ali, E. S., Hasan, M. K., Hassan, R., Saeed, R. A., Hassan, M. B., Islam, S., ... &Bevinakoppa, S. (2021). Machine learning technologies for secure vehicular communication in the internet of vehicles: recent advances and applications. *Security and Communication Networks*, 2021.
- [85] Daddanala, R., Mannava, V., Tawlbeh, L. A., & Al-Ramahi, M. (2021). Vehicle to Vehicle (V2V) Communication Protocol: Components, Benefits, Challenges, Safety and Machine Learning Applications. *arXiv preprint arXiv:2102.07306*.

- [86] Tang, F., Mao, B., Kato, N., &Gui, G. (2021). Comprehensive survey on machine learning in vehicular network: Technology, applications and challenges. *IEEE Communications Surveys & Tutorials*.
- [87] Talpur, A., &Gurusamy, M. (2021). Machine learning for security in vehicular networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*.
- [88] Balkus, S. V., Wang, H., Cornet, B. D., Mahabal, C., Ngo, H., & Fang, H. (2022). A Survey of Collaborative Machine Learning Using 5G Vehicular Communications. *IEEE Communications Surveys & Tutorials*.
- [89] Bustamante-Bello, R., García-Barba, A., Arce-Saenz, L. A., Curiel-Ramirez, L. A., Izquierdo-Reyes, J., & Ramirez-Mendoza, R. A. (2022). Visualizing Street Pavement Anomalies through Fog Computing V2I Networks and Machine Learning. *Sensors*, 22(2), 456.
- [90] Aznar-Poveda, J., Garcia-Sanchez, A. J., Egea-Lopez, E., & García-Haro, J. (2021). Simultaneous Data Rate and Transmission Power Adaptation in V2V Communications: A Deep Reinforcement Learning Approach. *IEEE Access*, 9, 122067-122081.
- [91] Choi, D., Yim, J., Baek, M., & Lee, S. (2021). Machine learning-based vehicle trajectory prediction using v2v communications and on-board sensors. *Electronics*, 10(4), 420.
- [92] Chbib, F., Zeadally, S., Khatoun, R., Khoukhi, L., Fahs, W., &Haydar, J. (2022). A secure cross-layer architecture for reactive routing in vehicle to vehicle (V2V) communications. *Vehicular Communications*, 38, 100541.
- [93] Kaur, G., &Kakkar, D. (2022). Hybrid optimization enabled trust-based secure routing with deep learning-based attack detection in VANET. *Ad Hoc Networks*, 136, 102961.
- [94] Paranjothi, A., &Atiquzzaman, M. (2022). A statistical approach for enhancing security in VANETs with efficient rogue node detection using fog computing. *Digital Communications and Networks*, 8(5), 814-824.
- [95] Cárdenas, L. L., León, J. P. A., &Mezher, A. M. (2022). GraTree: A gradient boosting decision tree based multimetric routing protocol for vehicular ad hoc networks. *Ad Hoc Networks*, 137, 102995.

- [96] Türkoğlu, M., Polat, H., Koçak, C., & Polat, O. (2022). Recognition of DDoS Attacks on SD-VANET Based on Combination of Hyperparameter Optimization and Feature Selection. *Expert Systems with Applications*, 117500.
- [97] Ahmad, S. A., & Shcherbakov, M. (2018, July). A survey on routing protocols in vehicular adhoc networks. In *2018 9th international conference on information, intelligence, systems and applications (IISA)* (pp. 1-8). IEEE.
- [98] Sami Oubbati, O., Chaib, N., Lakas, A., Bitam, S., & Lorenz, P. (2020). U2RV: UAV-assisted reactive routing protocol for VANETs. *International Journal of Communication Systems*, 33(10), e4104.
- [99] Canar, C. P. S., Yépez, J. J. T., & López, H. M. R. (2020). Performance of Reactive Routing Protocols DSR and AODV in Vehicular Ad-Hoc Networks Based on Quality of Service (Qos) Metrics. *Int. J. Eng. Adv. Technol*, 9(4), 2033-2039.
- [100] Kushwaha, U. S., Dixit, M. K., & Singh, A. K. (2022, January). AOMDV Intelligent Decision (AOMDV-ID) to Minimize Routing Delay for Diverse VANETs. In *2022 International Conference for Advancement in Technology (ICONAT)* (pp. 1-6). IEEE.
- [101] Faghiniya, M. J., Hosseini, S. M., & Tahmasebi, M. (2017). Security upgrade against RREQ flooding attack by using balance index on vehicular ad hoc network. *Wireless Networks*, 23(6), 1863-1874.
- [102] Reddy, V. A., Prasuna, P. S., Varu, R. D., Nikhila, P., & SriLakshmi, N. (2015). Performance analysis of DSDV, DSR routing protocols in vehicular Ad-hoc network (VANETs). *IJCSIT Int J Comput Sci Inf Technol*, 6, 2415-2418.
- [103] Clausen, T., & Herberg, U. (2010, June). Vulnerability analysis of the optimized link state routing protocol version 2 (OLSRv2). In *2010 IEEE International Conference on Wireless Communications, Networking and Information Security* (pp. 628-633). IEEE.
- [104] Muhtadi, A., Perdana, D., & Munadi, R. (2015). Performance evaluation of aodv, dsdv, and zrp using vehicular traffic load balancing scheme on vanets. *International Journal of Simulation, Systems, Science, and Technology, United Kingdom Simulation Society*.

- [105] Murthy, S., & Garcia-Luna-Aceves, J. J. (1996). An efficient routing protocol for wireless networks. *Mobile Networks and applications*, 1(2), 183-197.
- [106] Kang, I., Jeong, M. K., & Kong, D. (2012). A differentiated one-class classification method with applications to intrusion detection. *Expert Systems with Applications*, 39(4), 3899–3905. <https://doi.org/10.1016/j.eswa.2011.06.033>
- [107] Hu, W., Gao, J., Wang, Y., Wu, O., & Maybank, S. (2014). Online Adaboost-Based Parameterized Methods for Dynamic Distributed Network Intrusion Detection. *IEEE Transactions on Cybernetics*, 44(1), 66–82. <https://doi.org/10.1109/tcyb.2013.2247592>
- [108] Alom, Md. Z., Bontupalli, V., & Taha, T. M. (2015). Intrusion detection using deep belief networks. 2015 National Aerospace and Electronics Conference (NAECON). Dayton, 339–344. <https://doi.org/10.1109/naecon.2015.7443094>
- [109] Leandros, A., M. (2015). A Novel Distributed Intrusion Detection System for Vehicular Ad Hoc Networks. *International Journal of Advanced Computer Science and Applications*, 6(4). <https://doi.org/10.14569/ijacsa.2015.060414>
- [110] Aburomman, A. A., & Ibne Reaz, M. B. (2016). A novel SVM-kNN-PSO ensemble method for intrusion detection system. *Applied Soft Computing*, 38, 360–372. <https://doi.org/10.1016/j.asoc.2015.10.011>
- [111] Jabbar, M.; Aluvalu, R. (2017). RFAODE: A novel ensemble intrusion detection system. *Procedia Comput. Sci.*, 115, 226–234
- [112] Shone, N.; Ngoc, T.N.; Phai, V.D.; Shi, Q (2018). A deep learning approach to network intrusion detection. *IEEE Trans. Emerg. Top. Comput. Intell.*, 2, 41–50.
- [113] Liang, J., Chen, J., Zhu, Y., & Yu, R. (2019). A novel Intrusion Detection System for Vehicular Ad Hoc Networks (VANETs) based on differences of traffic flow and position. *Applied Soft Computing*, 75, 712–727. <https://doi.org/10.1016/j.asoc.2018.12.001>
- [114] Boukerche, A., & Wang, J. (2020). A performance modeling and analysis of a novel vehicular traffic flow prediction system using a hybrid machine learning-based model. *Ad Hoc Networks*, 106, 102224. <https://doi.org/10.1016/j.adhoc.2020.102224>

- [115] Batchu, R. K., & Seetha, H. (2021). A generalized machine learning model for DDoS attacks detection using hybrid feature selection and hyperparameter tuning. *Computer Networks*, 200, 108498. <https://doi.org/10.1016/j.comnet.2021.108498>
- [116] Türkoğlu, M., Polat, H., Koçak, C., & Polat, O. (2022). Recognition of DDoS attacks on SD-VANET based on combination of hyperparameter optimization and feature selection. *Expert Systems with Applications*, 203, 117500. <https://doi.org/10.1016/j.eswa.2022.117500>
- [117] Chen, Y., Lai, Y., Zhang, Z., Li, H., & Wang, Y. (2023). MDFD: A multi-source data fusion detection framework for Sybil attack detection in VANETs. *Computer Networks*, 224, 109608. <https://doi.org/10.1016/j.comnet.2023.109608>
- [118] Gyawali, S., & Qian, Y. (2019). Misbehavior Detection using Machine Learning in Vehicular Communication Networks. *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*. <https://doi.org/10.1109/icc.2019.8761300>

## LIST OF PUBLICATIONS

---

1. Khanna, H., & Sharma, M. (2021, July). A Packet Efficient Architecture for Vanet Based on AODV and Clustering. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 01-07). IEEE.
2. Khanna, H., Sharma, M., & Rattan, D. (2021). Secure and Authenticated Protocols for VANETs.
3. Khanna, H., & Sharma, M. (2022). An Improved Security Algorithm for VANET using Machine Learning. *Journal of Positive School Psychology*, 6(3), 7743-7756.
4. Sharma, M., & Khanna, H. (2018). Intelligent and Secure Vehicular Network using Machine Learning. *JETIR-International Journal of Emerging Technologies and Innovative Research* ([www.jetir.org](http://www.jetir.org)), ISSN, 2349-5162.