# TRUST FACTOR ANALYSIS IN AD – HOC NETWORKS

*Project report submitted in partial fulfillment of the requirement for the degree of*

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE AND ENGINEERING**

By

**Pulkita Vyas (121325)**

**Akansha Parashar (121326)**

UNDER THE GUIDANCE OF

**Dr. Shailendra Shukla**

to

Department of Computer Science & Engineering and Information Technology
**Jaypee University of Information Technology Waknaghat, Solan-173234**
**June, 2016**

# CERTIFICATE

I hereby declare that the work presented in this report entitled **“Trust Factor Analysis in Ad – Hoc Networks”** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2015 to December 2015 under the supervision of **Dr. Shailendra Shukla** (Assistant Professor, Computer Science ).
The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Pulkita Vyas
121325

Akansha Parashar
121326

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. Shailendra Shukla
Assistant Professor
Computer Science
Dated:

# ACKNOWLEDGEMENT

The Trust Factor Analysis in Ad - Hoc Networks marks the culmination of all the concepts assimilated while studying concepts of ad – hoc networks. It has presented us with an opportunity to use the technical know- how to create a real time system.

Learning the project under the guidance of our esteemed mentor Dr. Shailendra Shukla, who not only cleared all our ambiguities but also generated our high level of interest and gusto in the subject. We would also like to thank our Head of the Department, Brig. (Retd.) S.P. Ghrera for his underlying faith in the department of Computer Science and allocation of the project as well as its resources.

The prospect of working in a group with a high level of accountability fostered our spirit of teamwork and created a feeling of oneness which thus, expanded our ken, motivated us to perform to the best of our abilities and create a report of the highest quality.

To do the best quality work, with utmost sincerity and precision has been our constant endeavor.

Date : Pulkita Vyas (121325)

Akansha Parashar (121326)

# TABLE OF CONTENT

# LIST OF ABBREVIATIONS

1. DoD – Department of Defense
2. PRNET - Packet Radio Network
3. AP - Access Point
4. GSM – Global System for Mobile Communication
5. UMTS – Universal Mobile Telecommunications System
6. WLAN – Wireless Local Area Network
7. MANET - Mobile Ad hoc Network
8. VANET - Vehicular Ad hoc Network
9. DoS – Denial of Service

# LIST OF FIGURES

# LIST OF TABLES

# ABSTRACT

Trust Analysis in Ad - Hoc Networks allows the nodes to detect the nodes which are not secure for the transferring of the data packets. We aim to present a prototype system for real time tracking of a packet. The system uses a simple routing algorithm for the movement of packet.

The general requirements of a real time routing algorithm are that it should be computationally inexpensive and it should possess the ability to perform in different environments. This algorithm should be able to start and initialize itself with minimum knowledge about the environment.

The objective of this project is to calculate trust value of each node without knowing the infrastructure of the network. Ad hoc networks form quickly and this makes them indispensible in today's era. This algorithm can be used to secure the Vehicular Ad – Hoc Networks, Smart Phone Ad – Hoc Networks and many more.

# CHAPTER 1

# INTRODUCTION

## 1.1.History

The research on ad hoc networks was started in 1972 by the DoD - sponsored PRNET, which evolve into the Survivable Adaptive Radio Networks (SURAN) program. An ad hoc network is a collection of communication devices (nodes) that wish to communicate with each other, but have no fixed infrastructure available and have no previously determined organization of available links. Individual nodes are responsible for dynamically discovering all the other nodes with which they can directly communicate. Assumption is that not all nodes can directly communicate with each other, so nodes are required to spread packets on behalf of other nodes in order to deliver data across the network.

A significant feature of ad hoc networks is that it rapidly changes the connectivity and link characteristics are introduced due to node mobility and power control practices such as power consumption, throughput, routing, and connectivity. The mobile ad hoc network is at risk by its environment because of the vulnerabilities at channel and node level. The existing security mechanisms deal with only protecting resources from unauthorized access, but are not capable of protect the network from those who offer resources. Adding trust to the security infrastructures would improvise the security of these environments. In the proposed model, the trust value can be calculated to make decisions depending on their interaction of the nodes with each other about granting or rejecting the packets.

Such mechanisms are not only helpful in detecting malicious node, but also improve network performance because truthful nodes can avoid working with untrustworthy nodes. The focus of our work is to develop a framework that defines trust metrics and develop models of trust propagation in ad hoc networks. The proposed models are then applied to improve the performance of ad hoc network routing and to assist malicious node detection. The problem of defining trust metrics and trust relationship has been extensively studied for public key authentication, e-commerce and in P2P networks. Thus trust can be evaluated in many different ways.

## 1.2 Problem Statement

To study and analyze methods for calculating trust factor in the ever changing ad-hoc network and using this knowledge while transferring packets containing sensitive data.

## 1.3 Objectives and Scope of the Project

This project explores the field of computer networks with the broad aim of developing a system which is capable of sending data to trustable nodes .By calculating the trust factor we can recommend agents which are trust worthy in a mobile networks.

- To study trust factor in ad-hoc networks while sending the packets between the nodes.
- To design a real time system of the movement of nodes.
- Performance Requirements: computationally inexpensive, adaptability, self initializing and requires the knowledge of environment.
- The system would detect the movement of packet and calculate the trust factor of each node and give the trust value as output.

## 1.4 Methodology

Previous work on trust management in ad hoc networks focuses on trustworthiness evaluation process after initial trust relationship has been established. They do not, however, address how to obtain initial trust relationship partially because the meaning of the trust metrics is not clearly defined [1].The ad hoc network is at risk by its environment and various vulnerabilities that exist in the networks are :

- Channel vulnerability: Broadcasting through wireless channels can cause message eavesdropping and injection.
- Node vulnerability: Nodes lack physically protected places, thus they are susceptible to attacks.

- Absence of infrastructure: certification/ authentication authorities are missing.
- Dynamically changing network topology puts safety of routing protocols under threat.

Adding trust factor to the existing security infrastructures would enhance the security of these environments. There are two methods of calculating trust in networks:

(a)Trust propagation which is about predicting the trustworthiness of nonadjacent agents by combining the trust values through distinct indirect paths.

(b)Trust - based recommendation which is calculating trust on the basis of the recommendations provided by the neighboring nodes in the network.

# CHAPTER 2

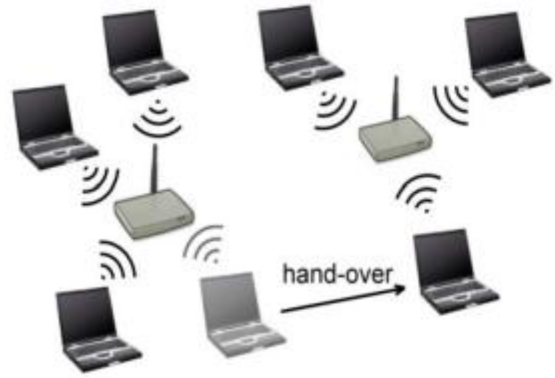
# LITERATURE SURVEY

The implementation of this project requires extensive knowledge of: -

1. Ad – Hoc networks
2. Trust Calculation
3. NS2 simulator v2.35

## 2.1 Ad – Hoc Networks

Ad-hoc network is an autonomous system node connected with wireless link (Corson et al., 1996).The node in the ad hoc network communicates with other node without any physical representation. The nodes in the ad hoc directly form the network whenever the communication is established in the network. Each node in the network communicates with other node using radio waves. The entire network is distributed and nodes are collaborated with each other without any fixed station or base station. An ad hoc network is local area network that builds an automatic connection to the nodes in network (Frodigh et al., 2000).



The wireless network architecture is to be classified in two ways, first one is infrastructure where the nodes are connected with their fixed physical representations. Thus, the nodes are communicated through AP. Examples for these kinds of wireless networks are GSM, UMTS and WLAN etc.

Second is infrastructure less, where the node is communicated without any fixed physical representation (Frodigh et al., 2000). The ad hoc networks are formed by connecting the terminals in the multi-hop distributed architecture (Stojmenovic and Lin, 2000). Due to the absence of centralized structure, the nodes in the ad hoc network acts as a router to send and receive data. Due to the non-static nature, ad hoc network avoid the single point of failure and make the network more robust .In ad hoc network, the transmission occurs between the source and the destination via intermediate nodes e.g. sensor.

### 2.1.1 Applications of ad hoc networks

In the research paper by Dr. Helen and D. Arivazhagan titled Applications, Advantages and Challenges of Ad Hoc Networks with the increased number of lightweight devices as well as evolution in wireless communication, the ad hoc networking technology is gaining effort with the increasing number of widespread applications. Ad hoc networking can be used anytime, anywhere with limited or no communication infrastructure. The previous infrastructure is fancy or annoying to use. The ad hoc network architecture can be used in real time business applications, corporate companies to increase the productivity and profit. The ad hoc networks can be classified according to their application as MANET which is a self-arranging infrastructure less network of mobile devices communicated through wireless link. VANET uses travelling cars as nodes in a network to create a mobile network. Wireless Sensor Network (WSN) consists of autonomous sensors which is used to control the environmental actions. The importance of ad hoc network has been highlighted in many fields which are described below:

Military area: An ad hoc networking will allow the military battleground to maintain an information of network among the soldiers, vehicles and headquarters.

Provincial level: Ad hoc networks can build an instant link between the multimedia network using notebook computers or palmtop computers to spread and share information among participants in the network.

Personal area network: A personal area network is a short range, localized network where nodes are usually associated within a given range.

Industry sector: In this Ad hoc network is widely used for commercial applications. It can also be used for emergency situation such as disaster help. The rapid development of non-existing infrastructure makes the ad hoc network easily to be used in emergency situation.

Bluetooth: Bluetooth can provide short range communication between the nodes such as a laptop and mobile phone or vice a versa.

### 2.1.2 Advantages of ad hoc networks

The rapidly development in ad hoc technology is widely used in portable computing such as laptop, mobile phone used to access the web services, telephone calls when the user is in travelling. Development of self-organizing network gives the advantage of decreasing the communication cost.

The advantages of an ad hoc network include:

- There is no central network administration.
- Nodes which are self-configuring are also routers.
- Self-healing is through continuous re-configuration.
- Scalability incorporates the addition of more nodes.
- Mobility allows ad hoc networks to create a fly in any situation where there are multiple wireless devices.

- Flexible ad hoc can be temporarily setup at anytime and in any place.
- Started costs get lowered due to decentralized administration.
- The nodes in ad hoc network need not to rely on any hardware and software. So, it can be connected and communicated quickly.

### 2.1.3 Challenges in ad hoc networking

The ad hoc networks are self-forming, self-maintaining, self-healing architecture. The challenges of this networks are, there is no fixed access point, dynamic network topology, different environment and irregular connectivity. Ad hoc network immediately forms and accommodate the modification and limit the power. Finally, ad hoc have no trusted centralized power. Due to the dynamic changing in property, the ad hoc faces challenges which are like

- Quality of Service (QoS)
- Scalability
- Security
- Power control
- Cooperation between nodes

## 2.2 Trust

There are several definitions which are given to trust in literature. Trust can be reflected by reliability, utility, availability, reputation, risk, confidence, quality of services and with other concepts.Trust is a relationship which is established between two entities for a specific action. In a particular, one entity trusts the other entity on behalf of the action performed by it. Trust is the critical link which is performed between observations (trust evidence) and the metrics that evaluate trustworthiness. In case of trust there are confusions in the definition of trust because in wired networks whether a node is reliable or not it is identified by certification of mechanism which is an indirect method of trust calculation. Without illustrating the meaning of trust, trustworthiness cannot be accurately determined from observations, and the calculation/policies/rules that govern the trust propagation cannot be justified.

In ad hoc networks, trust relationship can be established in two ways. The first way is through direct observations of other node’s behavior, such as dropping packets etc. The second way is through recommendations from other nodes.

Based on Yan Sun, Wei Yuy, Zhu Hany and K. J. Ray Liuy, "Trust Modeling and Evaluation in Ad Hoc Networks",2005 the linguistic descriptions of trust, decisions can be made based on linguistic trust policies or fuzzy logic. In some other schemes, discrete or continuous numerical values are assigned to measure the level of trust. For example, in, an entity’s opinion about the trustworthiness is described by a continuous value in [0*;* 1]. Triplet in [0*;* 1] is assigned to measure trustworthiness, and the elements in the triplet represent belief, disbelief, and uncertainty, respectively. In discrete, integer numbers are used. The basic idea of it is to build a trust model that provides a node with a device to evaluate the trust of its neighbors. A node assigns a trust level for each neighbor, which represents how trustworthy each neighbor is. Previous work on trust management in ad hoc networks focuses on trust evaluation process after initial trust relationship has been established. They do not, however, deal with how to obtain initial trust relationship partially because the meaning of the trust metrics is not clearly defined. In this paper, it is proposed that an information is theoretic framework of trust modeling and evaluation. In this framework, trust is a measure of uncertainty and it can be measured by entropy. From this understanding of trust, we developed axioms that address the basic rules for establishing trust through a third party (concatenation propagation) and through recommendations from multiple sources (multipath propagation).

Trust-oriented security framework, in order to secure the network from malicious nodes, can be used in making decisions for the following perspectives such as

Application execution

While the ad hoc network is in operation, numbers of applications like email, instant messaging, ftp and many others have to be started by the nodes in the network. As all of the participating nodes are ad hoc in nature so it is advisable to ensure the validity of the target node before starting any type of application execution as an interaction with the target node.

### Routing Environment

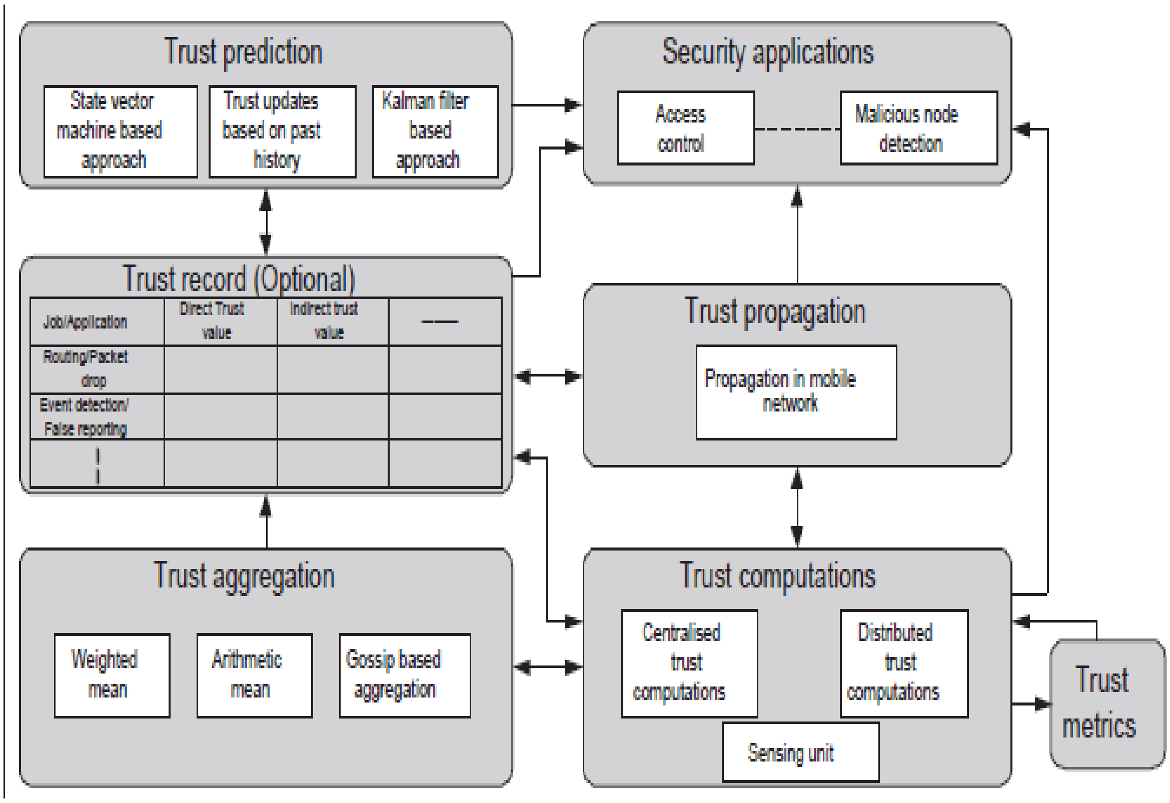
While the ad hoc network is in operation, there is a lot of packet flow over the network. The packet follows the path as per the routing protocol defined from node to node. In this context before forwarding the packet the source first gets the trust value of the receiver and is allowed to forward only if the trust value is above the threshold specified as per the policy. As the trust value is the result of past interactions so any misbehaving node can be barred by this validation on the basis of trust.

### Authentication

To accept or reject a public key certificate depends on the trust value of introducing node. Therefore the nodes involved in decision making is the value of trust that node has on the original.

### Pick the Best

Sometimes there is possibility the nodes have number of options i.e. number of nodes in the network, for an interaction or getting a service from it. In order to select among them, one of the criteria is to go ahead with the node for which the initiator has the highest trust value. So it leads to choosing the best among the available choices which increase security of the network.



| Job/Application | Direct Trust value | Indirect trust value | —— |
|---|---|---|---|
| Routing/Packet drop | | | |
| Event detection/ False reporting | | | |
| ¦ | | | |

Fig 1: Relationship among various trust blocks "Trust Computations and Trust Dynamics in Mobile Adhoc Networks: A Survey by Kannan Govindan and Prasant Mohapatra"

In Amandeep Verma and Manpreet Singh Gujral, Trust Oriented Security Framework for Ad hoc Network the trust model works as a trust service. This service is responsible for the trust evaluation, trust updation and trust propagation. The trust model encompassed the following components – Trust Configuration, Trust Assessment and Trust Appliance. The trust configuration engrosses-- characterizations of trust relationships, a range of trust categories, probable trust values. The trust assessment module is accountable for the trust evaluation. The trust appliance is used to supply trust values to the calling module. A trust value is a compute or quantification assigned by a source unit to its confidence in the trustworthiness of target unit. The trust value often signifies the view of a successful interaction, through which some desired outcome will be attained.

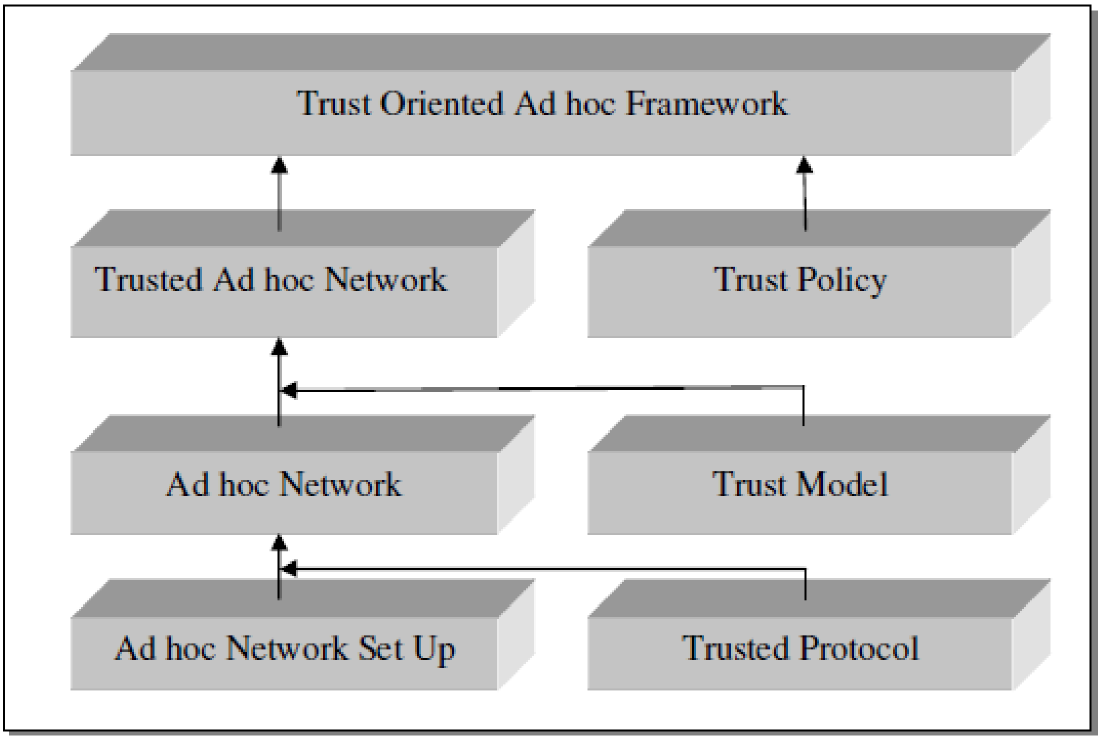


Fig 2: Abstract Representation of Trust oriented Ad hoc Network Framework ("Trust Oriented Security Framework for Ad Hoc Network" Amandeep Verma and Manpreet Singh Gujral)

Drawbacks of Existing Models:

- They do not deal with how to obtain initial trust relationship because the meaning of the trust metrics is not clearly defined.
- More power and time is taken to forward the recommendation.
- Memory requirements for storing recommendations have been ignored.
- In Maturity Based Model, the interaction of nodes is confined to their neighbors.
- Final trust value depends on the recommendations and not on the basis of individual ones.

### 2.3 NS2 simulator

Ns is a discrete event simulator which is under attack at networking research. Ns provides a substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks.

Ns began as a variant of the REAL network simulator in 1989 and has evolved substantially over the past few years. Ns has always included extensive contributions from other researchers, including wireless code from the UCB Daedelus and CMU Monarch projects and Sun Microsystems. It works at packet level and provides substantial support to simulate group of protocols like TCP, UDP, FTP, HTTP and DSR. It simulates both wired and wireless networks. It is primarily Unix based .In this TCL, scripting language is used. Ns-2 is a standard experiment environment in research community.

The ns-2 wireless simulation model simulates nodes moving in an unobstructed plane. Motion follows the random waypoint model [7], a node chooses a destination uniformly at random in the simulated region, chooses a velocity uniformly at random from a configurable range, and then moves to that destination at the chosen velocity. At the chosen waypoint, the node pauses for a configurable period before repeating the same process. In this model, the pause time acts as a substitute for the degree of mobility in a simulation; longer pause time amounts to more nodes being stationary for more of the simulation.

The two languages used in ns2 are:

**C++:** Requires systems programming language. Byte manipulation, packet processing, algorithm implementation are possible using this language. Run time speed is also important. Turnaround time (run simulation, find bug, fix bug, recompile, re-run) is slower.

**Tcl(Tool Command Language):** Simulation of a little unstable parameters or configurations. It is useful for quickly exploring a number of scenarios. The iteration time is more important.

**Network Animator:**

NAM is a Tcl/TK based animation tool which is used for viewing network simulation traces and real world packet traces. It supports topology layout, packet level animation, and various data inspection tools. It has a graphical interface, which can provide information such as number of packets drops at each link. The network animator

"NAM" began in 1990 as a simple tool for animating packet trace data. Nam began at LBL. It has evolved substantially over the past few years. The NAM development effort was an ongoing collaboration with the VINT project. It is used to show simple simulation of the ad-hoc network.We can start NAM with the command ‘nam<nam - file>’ where ‘<nam - file>’ is the name of a NAM trace file that was generated by NS or one can execute it directly out of the Tcl simulation script. The node movement can be seen using it.

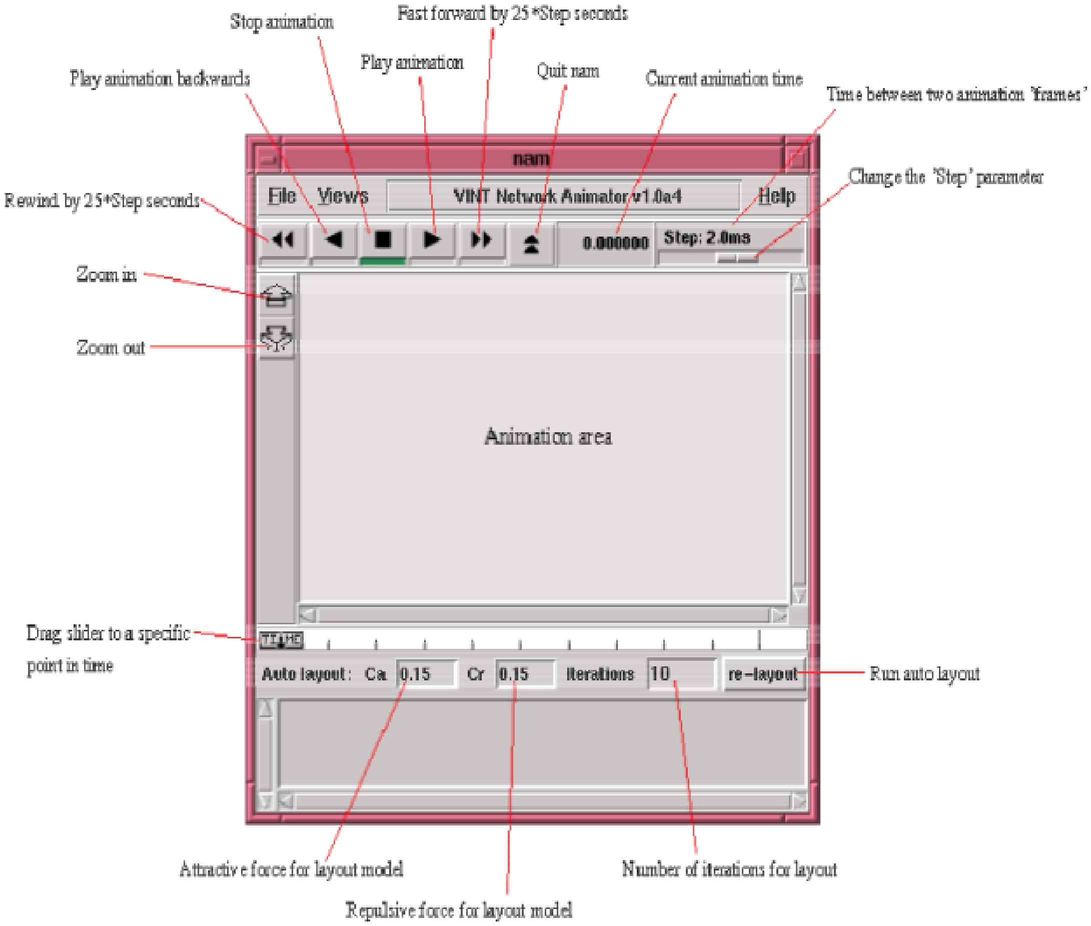

Stop animation
Fast forward by 25*Step seconds
Play animation
Quit nam
Current animation time
Play animation backwards
Time between two animation 'frames'
Change the 'Step' parameter
Rewind by 25*Step seconds
Zoom in
Zoom out
nam
File
Views
VINT Network Animator v1.0a4
Help
0.000000
Step: 2.0ms
Animation area
Drag slider to a specific point in time
TIME
Auto layout:
Ca
0.15
Cr
0.15
Iterations
10
re-layout
Run auto layout
Attractive force for layout model
Number of iterations for layout
Repulsive force for layout model

Fig 3: Network Animator

### X-graph

One part of the ns-allinone simulator is 'xgraph', a plotting program which can be used to create graphic representations of simulation results which is to be get from the simulation of the nodes.

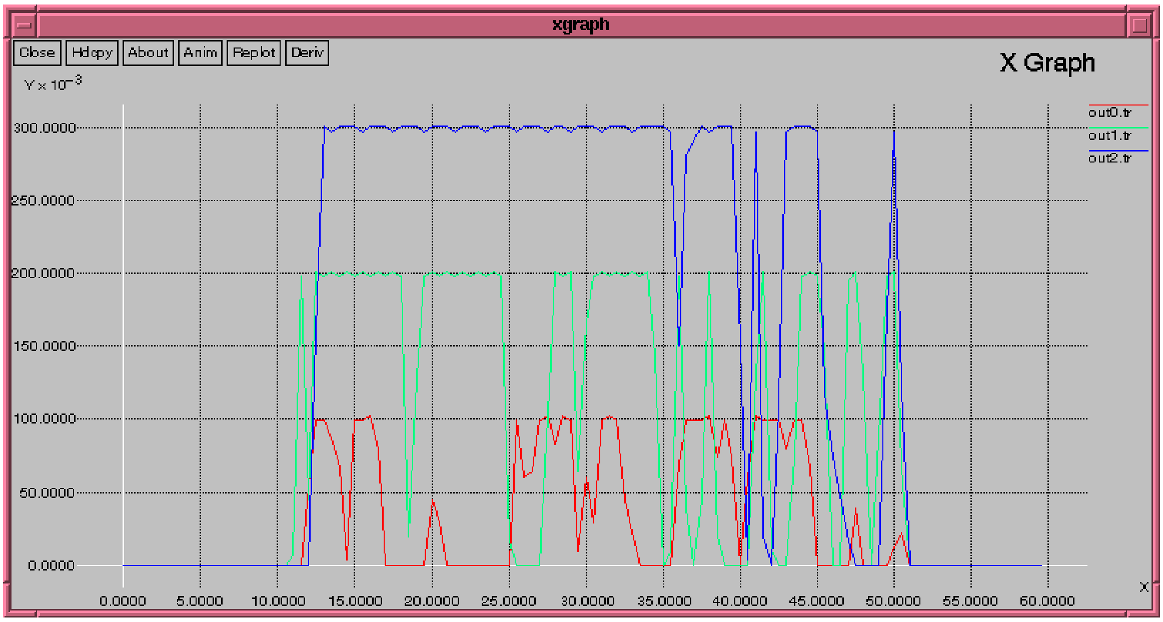


Fig 4: Xgraph

AODV Routing Protocol:

Ad hoc On-Demand Distance Vector (AODV) Routing is a routing protocol for the mobile ad hoc networks (MANETs) and also for other wireless networks. The Ad hoc On-Demand Distance Vector (AODV) algorithm enables dynamic, self-starting, multihop routing between the participating mobile nodes which are wishing to establish and maintain an network which is dynamic in nature. One distinguishing feature of AODV is that the usage of a destination sequence number for each route entry. The destination sequence number is created by the destination to be included along with any route information it sends to requesting nodes. Using destination sequence numbers loop freedom is to be ensured and it is simple to program. Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs) are the message types defined by AODV. These message types are received via UDP, and normal IP header processing applies.

DSDV Routing Protocol:

Destination-Sequenced Distance-Vector Routing (DSDV) is a table-driven routing protocol for ad hoc mobile networks which is based on the Bellman–Ford algorithm.

Every node will maintain a table which has a list of all the other nodes it has known either directly or through some of its neighbours. Every node has a single entry in the routing table. The entry table will have information about the node's IP address, last known sequence number and the hop count to reach that node. Along with these details the table also has to keep the track of the nexthop neighbor to reach the destination node, the timestamp of the last update received for that node.

The DSDV message consists of three fields, Destination Address, Sequence Number and Hop Count.

Each node uses 2 mechanisms to send out the DSDV updates which are,

1. Periodic Updates

   Periodic updates are sent out after every m_periodicUpdateInterval. In this update the node broadcasts out its entire routing table.

2. Trigger Updates

   Trigger Updates are small updates in between the periodic updates. These updates are sent out whenever a node receives a DSDV packet that caused a change in its routing table. The original paper did not clearly mention when for what change in the table should a DSDV update be sent out. The current implementation sends out an update irrespective of the change in the routing table.

   The updates are accepted on the basis of metric for a particular node. The first factor that determines the acceptance of an update is the sequence number. It has to accept the update if the sequence number of the update message is higher irrespective of the metric. If the update with same sequence number is received, then the update with least metric (hopCount) is given precedence.

Python:

Python is a language which is widely used as a high-level, general purpose, interpreted, dynamic programming language. Its design philosophy is to highlight code readability, and its syntax which allows programmers to express concepts in a fewer lines of code rather than in C++ or Java The language provides constructs planned to enable clear programs on both a small and large scale.

Python supports multiple programming paradigms, which includes object-oriented, imperative and functional programming or procedural styles. It features on a dynamic type system and automatic memory management and has a large and standard library.

Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems. Using third-party tools, such as Py2exe or Pyinstaller, Python code can be packaged into stand-alone executable programs for some of the most popular operating systems, so Python-based software can be distributed to, and used on, So there is no need to install a Python interpreter.

# CHAPTER 3

# SYSTEM DEVELOPMENT

The overall design is as depicted in figure below. In ad – hoc networks nodes have some communication range so as the diagram depicts the node first sees its communication range and then the trust factor. In the flow chart if the value of trust is in the acceptable range then deliver the packet else assign a new action to all the nodes present in the neighboring area. Observe the nodes and the recommendations from other nodes then calculate the trust factor and send feedback accordingly. This process is repeated until trust values for all nodes are calculated and for the time interval in which the network remains unchanged.

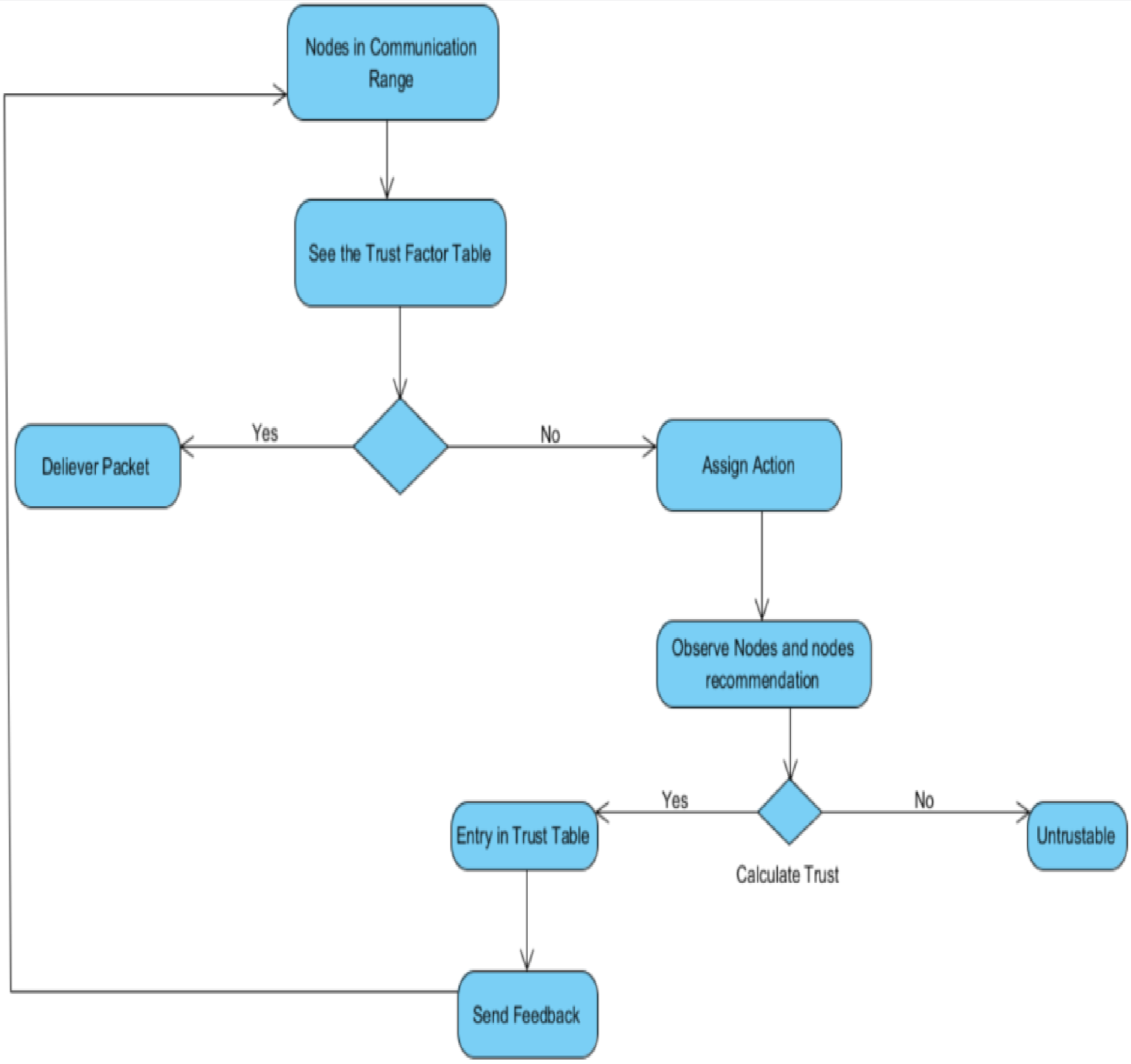


Fig 5: Block Diagram for System Flow

### 3.1 Modular Description

According to the findings in “Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity-Based Model by Pedro B. Velloso, Rafael P. Laufer, Daniel de O. Cunha, Otto Carlos M. B. Duarte, and Guy Pujolle” the system compromises of the following modules:

The model can be divided in two different plans.

The responsibility of the Learning plan is gathering and converting the information into knowledge about the network. For example, the Learning plan monitors the behavior of each neighboring node.

The Trust plan defines how to evaluate the level of trust for each neighbor using the information provided by the Learning plan and then the information is exchanged with the neighboring nodes.

Both these plans i.e. the Learning and the Trust Plan can interact with all layers of the TCP/IP model. Therefore, the learning process takes into account all the information from the various layers and the trust values generated by the Trust plan are also available for all layers.

Since not only malicious nodes are taken into consideration but also self-seeking behaviors of the nodes due to the resource constraints, the trust value of a particular node is associated with a specific parameter, like the packets that are forwarding data, sending recommendations, and other scopes for application. Therefore, it is considered that a node might behave in a way that is different according to the scope and the resource constraints for that network. As a result, the type of information that is to be collected by the Learning plan depends on the defined scopes. For example, in the routing process, the Learning plan must monitor if neighboring nodes are responding to the requests for route or if they respond by sending false routes.

The Learning plan relies on three basic components. The Behavior Monitor observes neighbors in order to collect information about their behavior. It must be able to notice other nodes’ actions and transmit them to the Classifier. The Behavior Monitor also indicates the presence of new neighbors to the Recommendation Manager. The Classifier is the component dedicated to reason about the information collected by the Monitor. The Classifier decides the quality of an action according to a previously defined classification. The Classifier then sends its verdict to the Experience Calculator. Finally, the Experience Calculator estimates a partial trust value for a given node based on the information received by the Classifier. The focus is on the Trust plan and it is assumed that the Learning plan is not sufficient, which only sees a part of the behavior of other nodes. The Trust plan is made of five components. Each node keeps a Trust Table which has the trust level for each of the neighboring nodes. In addition to this, a node can also save the opinion of its neighboring nodes about their common neighbors on the Trust Table. Each entry on the Trust Table is coupled with a timeout. Therefore, an entry is removed from the Trust Table whenever the node associated to that entry is no longer a neighboring node or when it has expired. All the recommendations related to that entry are removed. In this model, nodes can also store an additional table that is not compulsory called the Auxiliary Trust Table (ATT) which has the variance of each trust level and for how long they keep that information. The aim of the Auxiliary Trust Table is to provide the nodes with additional information that improves the trust factor assessment. But this requires more energy consumption and nodes with power or storage constraints can choose not to create this additional table.

In order to tackle the ever changing characteristics of the ad hoc network, it has three operation modes:

- Simple

  Nodes with low power/storage capacity operate in the simple mode, in which they use just the main Trust Table and the Recommendation Exchange Protocol (REP) protocol is optional.

- Intermediate

  Nodes with a medium capacity operate in the intermediate mode, which also keeps the recommendations of other nodes.

- Advanced

  In the advanced mode, nodes implement the whole trust system with all features.

It is considered that nodes operate in the advanced mode. The Recommendation Manager is responsible for receiving, sending, and storing recommendations. The interactions between the Network Interface and the Recommendation Manager are performed by the Recommendation Exchange Protocol (REP). The reception of a recommendation involves two actions. First, the recommendation is stored in the Auxiliary Trust Table (ATT) and then it is forwarded to the Recommendation Calculator component. The Recommendation Calculator computes all the recommendations for a given neighboring node and determines a trust value based on the opinions of other nodes. This value is passed to the Trust Calculator component. The Trust Calculator evaluates the trust factor based on the trust values received from the Experience Calculator i.e. their individual experiences and the Recommendation Calculator based on the recommendations of the neighboring nodes. The Trust Calculator also informs the Recommendation Manager the need for sending a trust recommendation. The proposition is that the system only requires interactions with the neighboring nodes and only stores information about them. This is an important for mobile ad hoc networks which are made by portable devices that have energy, processing, and memory restrictions.

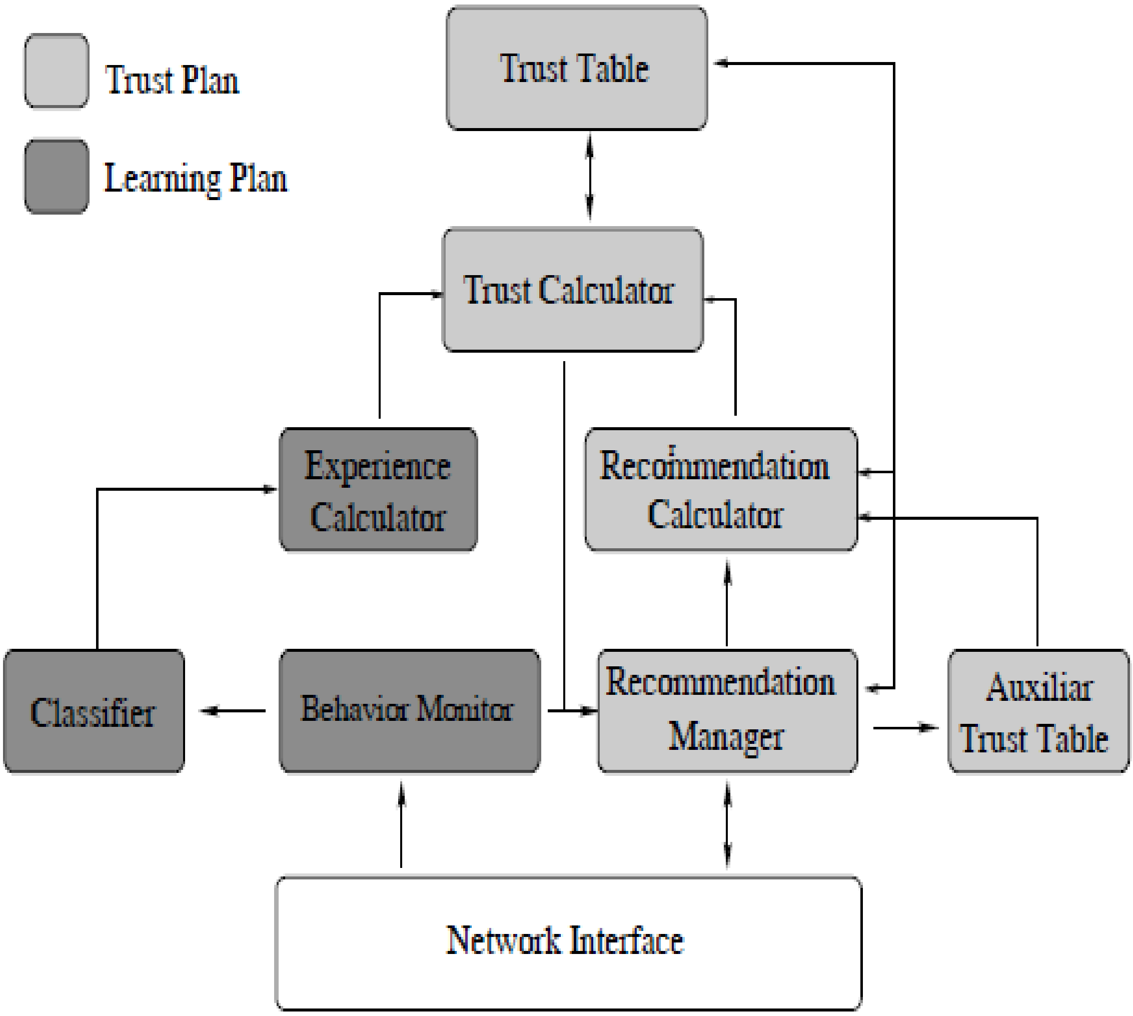


Fig 6: The proposed trust system components "Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity-Based Model Pedro B. Velloso, Rafael P. Laufer, Daniel de O. Cunha, Otto Carlos M. B. Duarte, and Guy Pujolle"

## 3.2 System Architecture

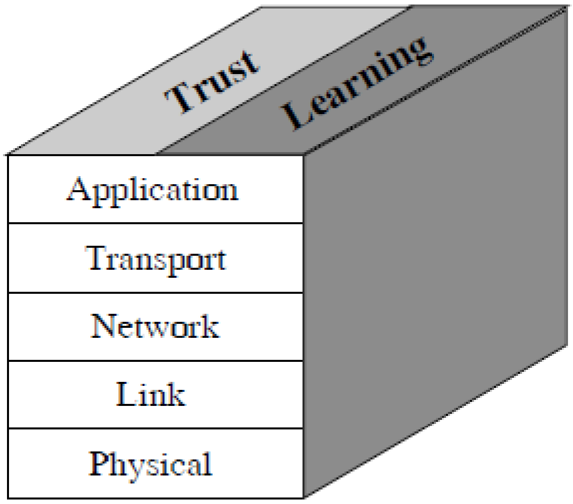


Fig 7: The proposed trust model architecture "Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity-Based Model Pedro B. Velloso, Rafael P. Laufer, Daniel de O. Cunha, Otto Carlos M. B. Duarte, and Guy Pujolle"

## 3.3 Algorithms for Model Development

### 3.3.1 Trust-based Recommendation based on Graph Similarity

According to "Trust-Based Recommendation Based on Graph Similarity by Chung-Wei Hang and Munindar P. Singh" a trust network is a graph where nodes represent agents and edges represent trust relations.

A trust relation from agent 'u' to agent 'v' shows how much trust 'u' has in 'v'. So, an edge in a trust network is associated with a trust value as its weight. Depending on the trust models, a trust value can be a single scalar, a Beta distribution, or follow another representation. The trust relations can be obtained from a direct communication or from a recommendation via trust propagation.

For example; a social network such as Facebook is a network where all edges are modeled and have some trust values. A trust value as a single scalar.

### 3.3.2 Trust Management using a Scalable Maturity-Based Model

According to "Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity-Based Model Pedro B. Velloso, Rafael P. Laufer, Daniel de O. Cunha, Otto Carlos M. B. Duarte, and Guy Pujolle" trust is evaluated using:

A. Trust level evaluation

In this we define the trust level of evaluation from node $a$ about node $b$, as a weighted sum of its own trust (monitor) and the recommendations of this through neighbors. The fundamental equation is

$$T_a(b) = (1-\alpha)Q_a(b) + \alpha R_a(b) \qquad (1)$$

here the variable $Qa(b)$, that ranges from [0,1], represents the trust of node $a$ has on node $b$. It is based only on its own observations and ($b$), that ranges from [0,1], is the aggregate value of the recommendations from all other neighbors. The variable $\alpha$, that ranges from [0, 1], is a parameter in the model that allows nodes to choose the most relevant factor. In our model, the value of ($b$) is given by

$$Q_a(b) = \beta E_a(b) + (1-\beta)T_a(b), \qquad (2)$$

where $Ea$ represents the trust value obtained by the decision of the actions which is performed by the neighbor, and the variable $Ta(b)$ gives the last trust level value which is stored in the Trust Table. The variable $\beta$, that ranges from [0, 1], allows different weights for the factors of the equation, selecting which factor is the more relevant at a given moment of time. Equations 1 and 2 describe how the Trust is Calculated by combining the information from the Experience Calculator (($b$)), the Recommendation Calculator (($b$)), and the Trust Table (($b$)) to derive a trust level.

B. Recommendation computation

The trust level is calculated by considering the recommendations of neighbors obtained by the Recommendation Exchange Protocol (REP). Equation 1, represents the aggregate trust that the neighbors of node $a$ have on node $b$. First, node $a$ defines a set $Ka$ which is a subset of its neighbors that comprises all nodes whose trust level is above a certain level of threshold, to increase the confidence of recommendations.

The recommendation, ($b$), is defined as the weighted average of the recommendations from all the nodes $i \in Ka$ about node $b$. The weight for a recommendation from a neighbor $i$ is the trust level that node $a$ has on node $i$, as follows:

$$R_a(b) = \frac{\sum_{i \in K_a} T_a(i) M_i(b) X_i(b)}{\sum_{j \in K_a} T_a(j) M_j(b)}. \tag{3}$$

The recommendations consider not only the trust level of other nodes ($Ta$), but also the accuracy ($Xi$) and the relationship maturity ($Mi$). The accuracy of a trust level is based on the standard deviation. The value in the Trust Table of node $a$ regarding node $b$ is associated with a standard deviation $\sigma a(b)$, which refers to the variations of the trust level that node $a$ has observed about node $b$. We use $X$ as a random variable with a normal distribution to represent the uncertainty of the recommendation. It can be expressed as

$$X_i(b) = N(T_i(b), \sigma_i(b)). \tag{4}$$

The recommendation of node $i$ about node $b$ is weighted by ($b$), which defines the maturity of the relationship between nodes $i$ and $b$, measured at node $i$. The relationship maturity is a measure of the time that two nodes have known each other. We use the relationship maturity to give more significance to the nodes that know the evaluated neighbor for a longer time. Accordingly, we assume that the trust level of a older neighbor has already converged to a common value within the network and therefore its opinion should be more applicable than the opinion of a new neighbor.

### C. The First Trust Assignment

We divide the trust scheme in two distinct phases. In the initial phase, nodes first meet simply and then assign a trust level to each other. The second phase is the trust level update, which assumes that the nodes have already met each other. When a node first meets a specific neighbor, it assigns an initial level of trust to this neighbor. We classify the first trust assignment strategy as discreet or optimistic. In the discreet strategy the node does not trust the strangers and considers that every new neighbor is threat to the network. As a consequence, the node assigns a low value of trust for the new neighbor. On the other hand, the optimistic strategy assumes that every node is reliable until it is proven . In such case, the node associates a high level of trust for new neighbors.

Right in the middle of these two strategies, one could think of a moderate strategy, in which the node assigns an intermediate level of trust for strangers.

The first trust assignment can also take into account the recommendation of known neighbors weighted by their trust levels. For a node *a* to calculate the first trust level of a node *b*, use the same approach as Equation 1, but replacing the term that reflects its own experience by the First Trust Value, (*Fa*), given by:

$$T_a(b) = (1 - \alpha)F_a + \alpha R_a(b), \tag{5}$$

where *Fa* is the value used by node *a* according to the adopted strategy, *Ra*(*b*) is the aggregate recommendation of neighbors about node *b*, and *α* is the weight factor that allows us to give more relevance to the desired parameter.

D. The Recommendation Exchange Protocol

The recommendation from a node *i* ∈ *Ka* includes the trust level (*b*) of the target node *b*, its accuracy (*b*) and for how long they know each other, (*b*). For a node that does not implement the secondary Trust Table the recommendation includes just the trust level *Ti*(*b*).Use Recommendation Exchange Protocol (REP) as a part of the Recommender Manager .This protocol allows nodes to exchange recommendations among them and only considers interactions with neighbors, which significantly simplifies the protocol. Thus, all messages are transmitted by one hop and broadcasts by avoiding flooding in multihop communications. When using IP to broadcast the message, the Time to Live (TTL) field is set to 1. The protocol is composed of three messages: Trust Request (TREQ) message, Trust Reply (TREP) message, and Trust Advertisement (TA) message. When nodes first meet, each one broadcasts a Trust Request (TREQ) message to their neighbors with the IP address of the new neighbor as the target node. All neighbors receive the TREQ message and check if the target node is a neighbor or not. Nodes that have the target node as a neighbor, will answer with a

Trust Reply (TREP) message, which contains the recommendation about the target node.

### 3.3.3 Securing Ad Hoc Network with Enhanced Trust Calculation Method

According to the findings in the paper “A Novel Approach for Securing Mobile Ad Hoc Network with an Enhanced Trust Calculation Method” by Amit Chauhan, Prof. R.P. Mahajan

There are two methods for the calculation of trust value for the nodes.

Direct Trust Value:

This method is for the direct trust relationships, where a mobile node in a range can observe the activities of its neighbors carefully and calculate the value of trust on its own without external help.

Recommender Trust Value:

This method is for recommender trust relationship. In this case, trust value of out of range node is requested using RRQ (Recommendation Request).

A malicious activity by any node can be detected and other nodes are informed using Single Intrusion Detection (SID). It is possible that a malicious node may broadcast a false SID against a genuine node or due to the unavoidable conditions like poor connectivity, error in received packets, etc. a node maybe classified as a compromised node by its neighboring nodes and an SID may broadcasted against the node. Thus, instead of accepting the SID completely without checking, the following parameters are taken into consideration by a node receiving SID broadcast.

Algorithm to validate SID

- Trust level of a node, which is broadcasting SID must be checked against a compromised node.
- If a compromised node is in the range of the node, it will watch a compromised node for a certain period.
- It will request other neighboring nodes for their recommendations about the compromised node.
- Depending on its conclusion, a node maybe assigned a new trust value.

### 3.3.4 Trust Modeling and Evaluation in Ad – Hoc Networks

According to “Trust Modeling and Evaluation in Ad Hoc Networks” by Yan Sun, Wei Yu, Zhu Han, and K. J. Ray Liu to establish trust relationship and to measure trust there are three cases:

- When subject believes that agent will perform action for sure.
- When the subject believes that agent will not perform action for sure.
- When the subject has no idea about the agent at all.

Trust is not necessarily symmetric nor is it transitive. For the Trust metrics it gives trust value of ‘1’ when subject believes that agent will surely perform the action .In the second case the trust value ‘-1’ is assigned when the subject is sure that agent will not perform action, and ‘0’ in the third case when the subject has no idea about the agent at all.

Let T {subject: agent; action} denote the trust value of the trust relationship

{subject : *agent;* action}, and P{subject : agent; action} denote the probability that the agent will perform the action in the subject’s point of view. The entropy-based trust value is calculated as follows:

T{subject : agent; action}= $1 - H(p)$; for $0.5 \leq p \leq 1$;

$$H(p) \quad 1; \text{ for } 0 \leq p < 0.5; \tag{1}$$

where $H(p) = -plog_2(p) - (1 - p)\, log_2(1 - p)$ is the entropy function and

$p$ = P{subject : agent; action}

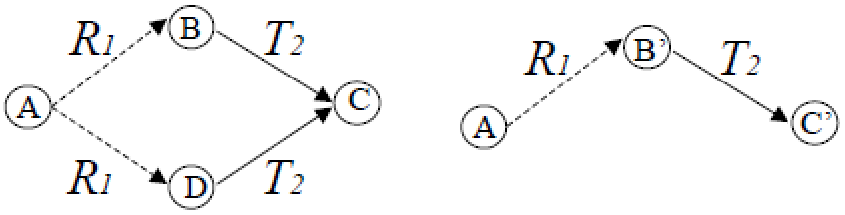


Fig 8: Combing Trust Recommendations “Trust Modeling and Evaluation in Ad Hoc Networks” by Yan Sun, Wei Yu, Zhu Han, and K. J. Ray Liu

Assuming that *A* and *B* have established $\{A : B; action_r$, and *B* and *C* have established

$\{B : C; action\}$. Then, $\{A : C; action\}$ can be established if

(1) $action_r$ makes recommendation of other nodes about performing *action*.

(2) The trust value of $\{A : B; action_r\}$ is positive.

The first condition is necessary because the nodes that perform the action may not make correct recommendations. The second condition is essential because untrustworthy node can give a recommendation that can be totally unrelated to the truth. Thus, the best way is not to take recommendations from untrustworthy nodes.

### 3.2 Proposed Model

In this we calculate trust value for all the nodes in the network. Each node's trust factor is calculated by using the delivery of packets, received, forwarded and dropped information. In this two routing protocols is used which are AODV and DSDV. Each node's trust value can be calculated on the basis of the packets forwarded, dropped, received and sent. We calculate trust value by the formula:

$$\frac{Sent\ -\ Dropped}{Received\ -\ Forwarded}$$

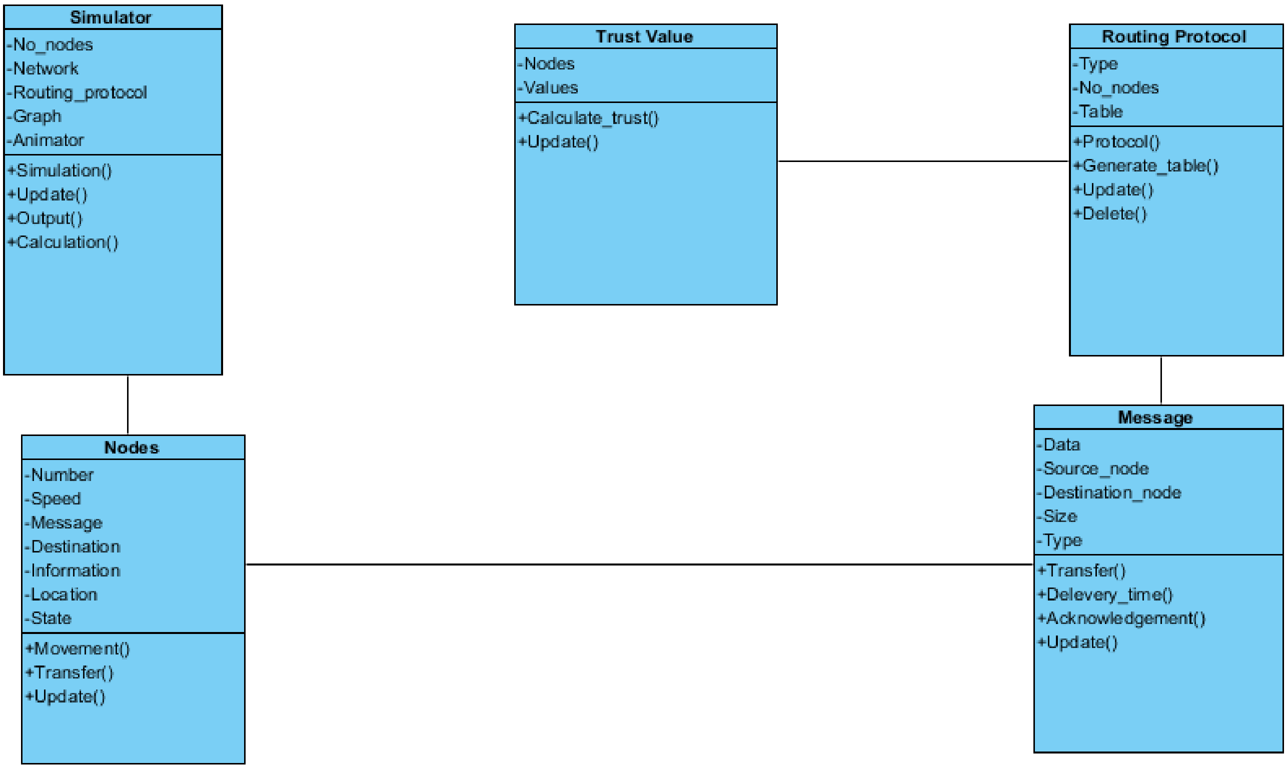


Fig 9: Class Diagram

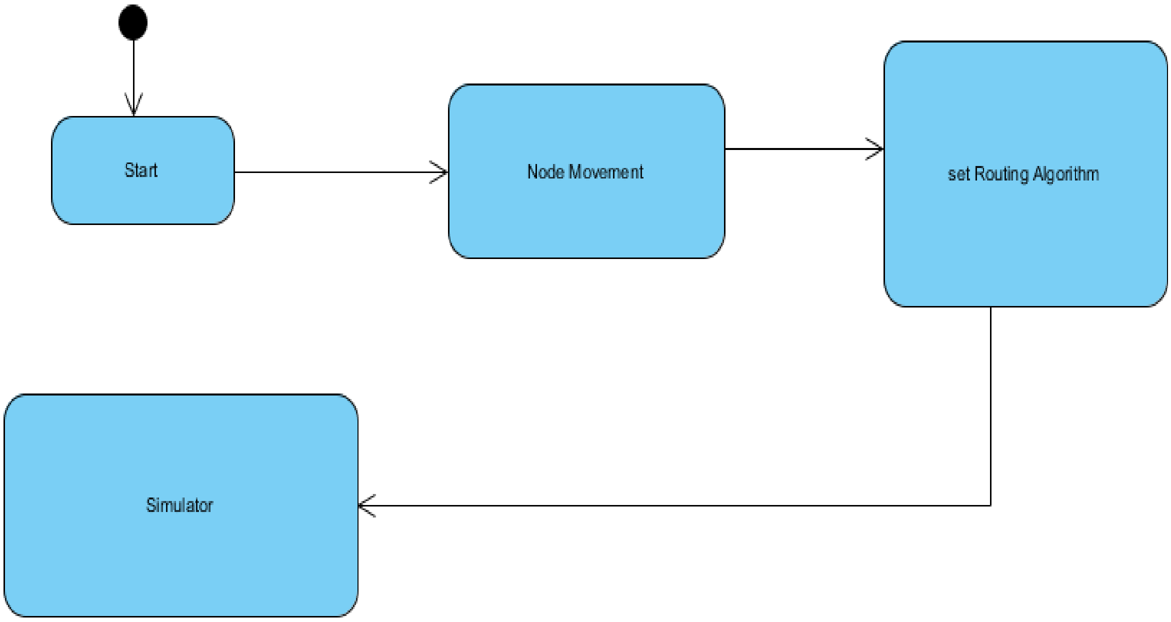


Fig 10: State Transition Diagram

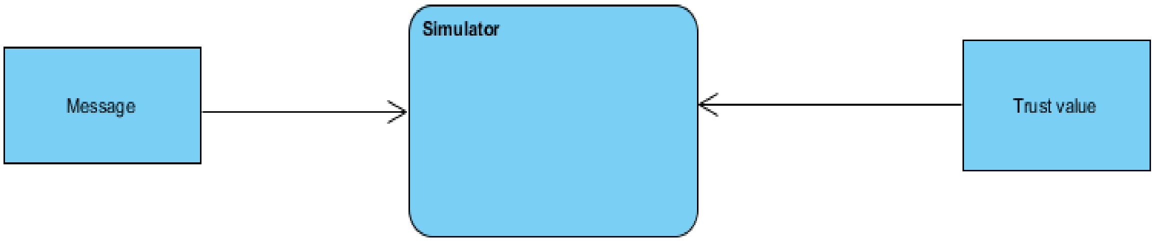


Fig 11: Data Flow Diagram (Level – 0)

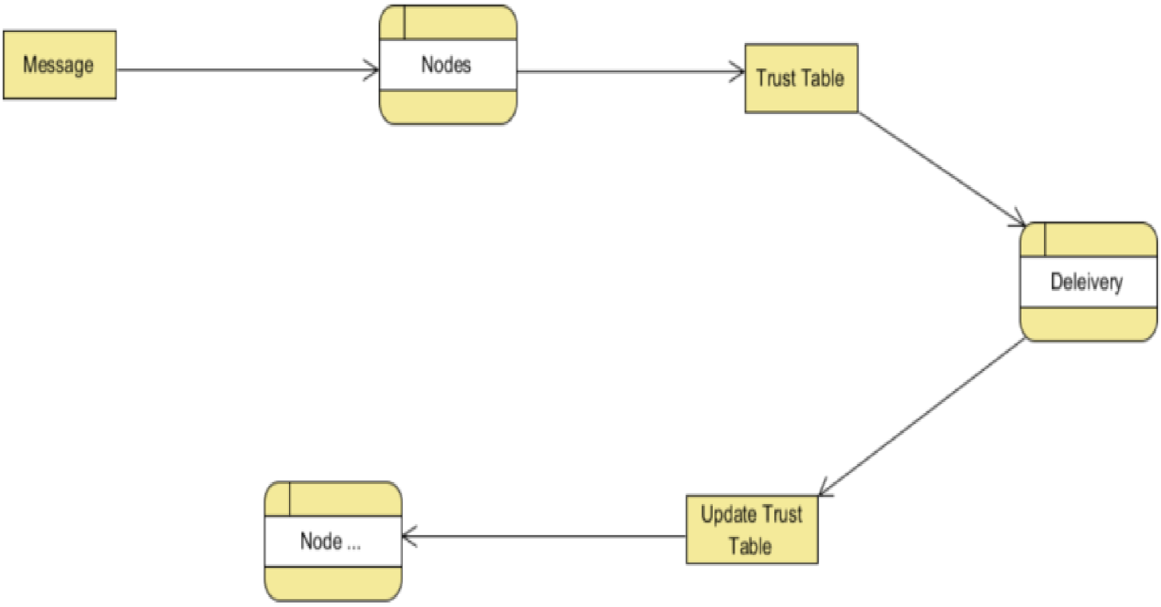


Fig 12: Data Flow Diagram (Level – 1)

# CHAPTER 4

# PERFORMANCE ANALYSIS

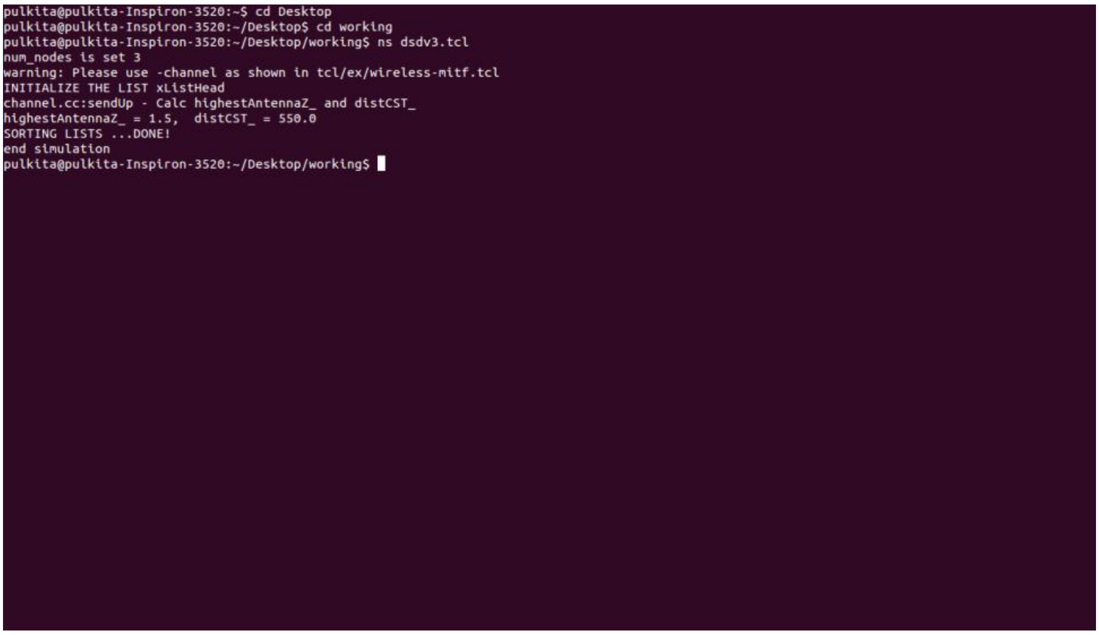
The environment for all simulations:

- Ubuntu 14.04
- NS2 Version 2.35

Languages Used:

- OTCL
- Python

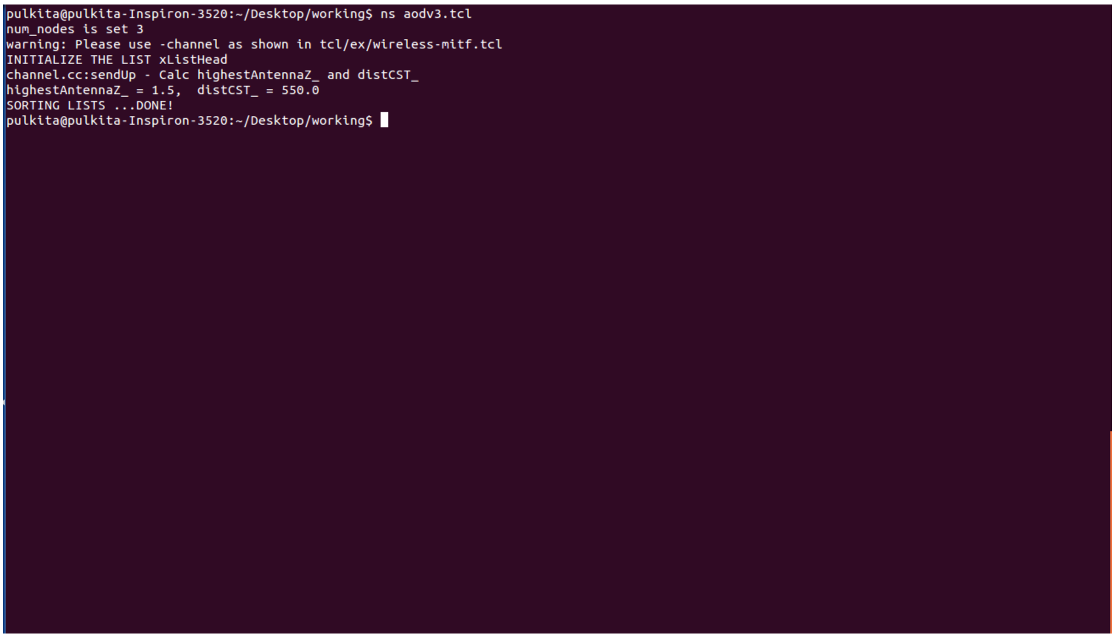
## 4.1 Simulation using DSDV Routing Protocol



```
pulkita@pulkita-Inspiron-3520:~$ cd Desktop
pulkita@pulkita-Inspiron-3520:~/Desktop$ cd working
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ ns dsdv3.tcl
num_nodes is set 3
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
end simulation
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ 
```

Fig 13: Simulation of DSDV Protocol

## 4.2 Simulation using AODV Routing Protocol



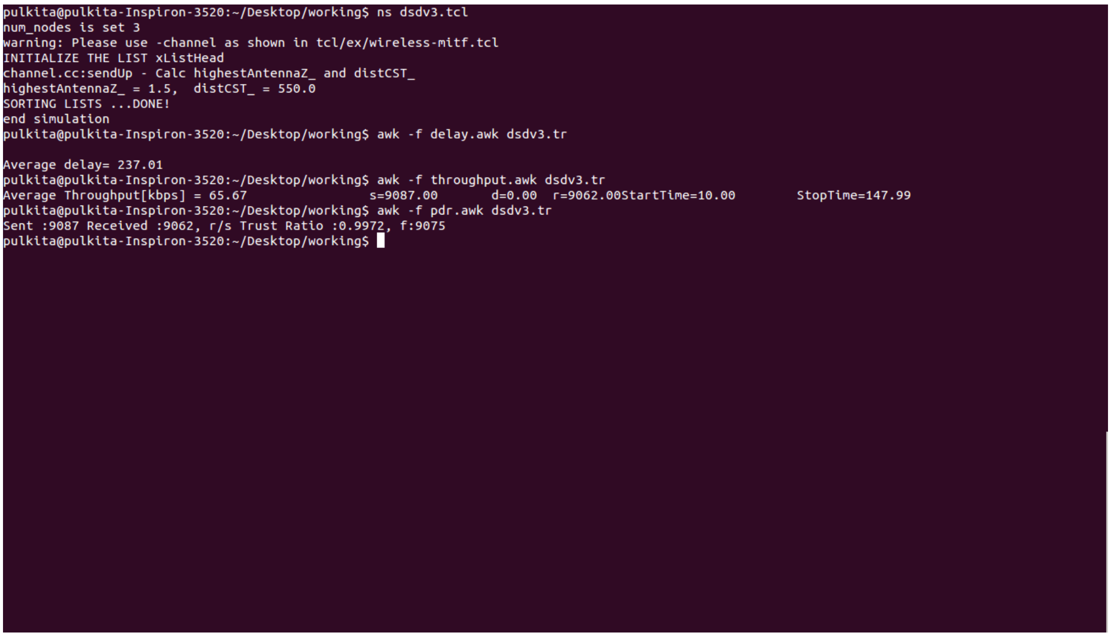
```
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ ns aodv3.tcl
num_nodes is set 3
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ 
```

Fig 14: Simulation of AODV Protocol

## 4.3 Calculation of Trust Factor using DSDV Routing Protocol

We calculate the trust factor by using DSDV as routing protocol. The formula used is:

$$\frac{Packets\ received\ successfully}{Total\ number\ of\ packets}$$



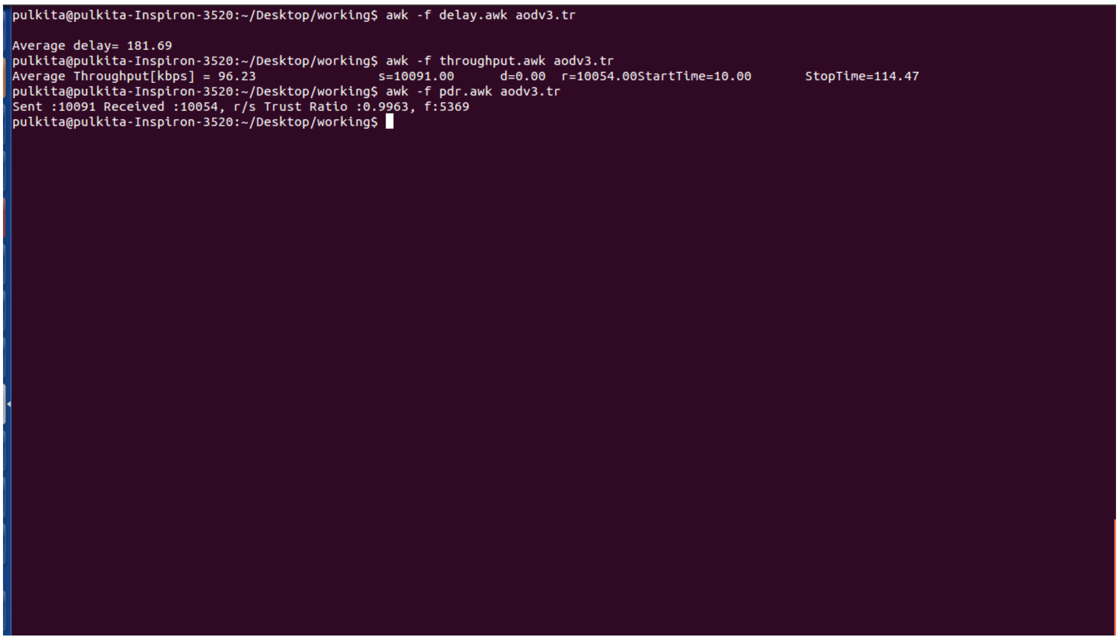
```
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ ns dsdv3.tcl
num_nodes is set 3
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
end simulation
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f delay.awk dsdv3.tr

Average delay= 237.01
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f throughput.awk dsdv3.tr
Average Throughput[kbps] = 65.67                s=9087.00       d=0.00  r=9062.00StartTime=10.00        StopTime=147.99
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f pdr.awk dsdv3.tr
Sent :9087 Received :9062, r/s Trust Ratio :0.9972, f:9075
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ 
```

Fig15: Calculation of delay, throughput and trust ratio

## 4.4 Calculation of Trust Factor using AODV Routing Protocol

We calculate the trust factor by using AODV as routing protocol. The formula we used is:

$$\frac{Packets\ received\ successfully}{Total\ number\ of\ packets}$$



```
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f delay.awk aodv3.tr

Average delay= 181.69
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f throughput.awk aodv3.tr
Average Throughput[kbps] = 96.23                s=10091.00      d=0.00  r=10054.00StartTime=10.00       StopTime=114.47
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f pdr.awk aodv3.tr
Sent :10091 Received :10054, r/s Trust Ratio :0.9963, f:5369
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ 
```

Fig 16: Calculation of delay, throughput and trust ratio

## 4.5 Comparison between AODV and DSDV Routing Protocols

Delay: The delay of a network specifies how long it takes for a bit of data to travel across the network from one node or endpoint to another. It is typically measured in multiples or fractions of seconds. Delay may differ slightly, depending on the location of the specific pair of communicating nodes.

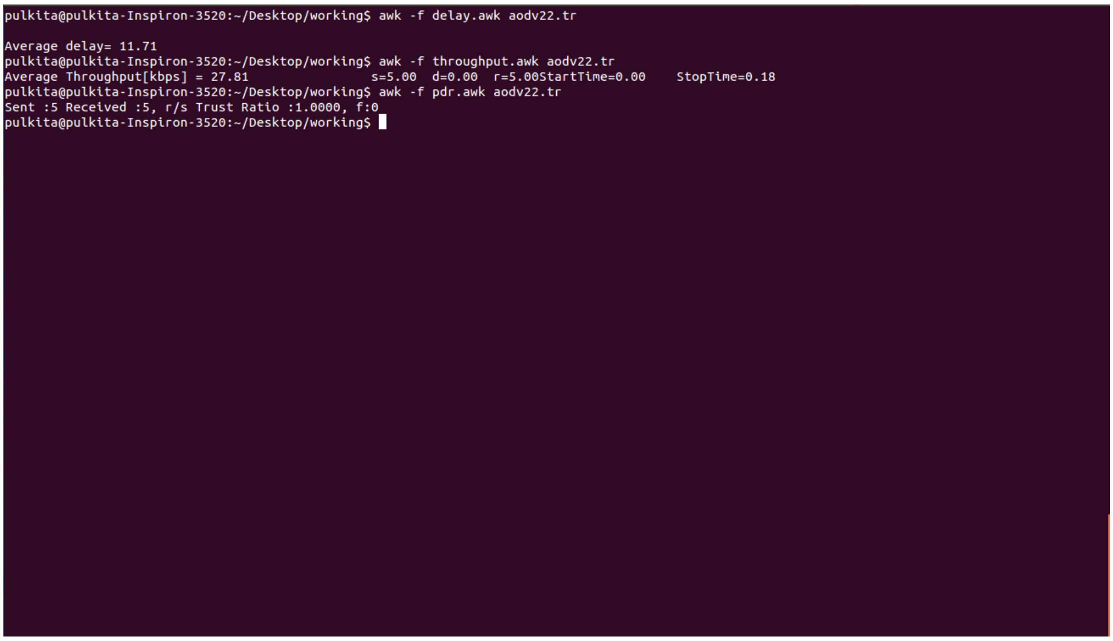
Throughput: Throughput is the rate of successful message delivery over a communication channel. The data these messages belong to may be delivered over a physical or logical link or it can pass through a certain network node. Throughput is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second (p/s or pps) or data packets per time slot.

Trust Ratio: The Trust Ratio for the entire network is fraction of the packets that were successfully received from the total number of packets that were sent during a course of time.

| | DSDV Protocol | AODV Protocol |
|---|---|---|
| Delay | 237.01 | 181.69 |
| Throughput | 65.67 | 96.23 |
| Trust Ratio | 0.99 | 0.99 |

Table 1: Comparison between AODV and DSDV

## 4.6 Calculation of trust factor of each node at time (t)



```
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f delay.awk aodv22.tr

Average delay= 11.71
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f throughput.awk aodv22.tr
Average Throughput[kbps] = 27.81                s=5.00  d=0.00  r=5.00StartTime=0.00    StopTime=0.18
pulkita@pulkita-Inspiron-3520:~/Desktop/working$ awk -f pdr.awk aodv22.tr
Sent :5 Received :5, r/s Trust Ratio :1.0000, f:0
pulkita@pulkita-Inspiron-3520:~/Desktop/working$
```

Fig 17: Calculation of delay, throughput and trust ratio

We calculate each node’s trust factor by using the formula:

$$\frac{Sent - Dropped}{Received - Forwarded}$$

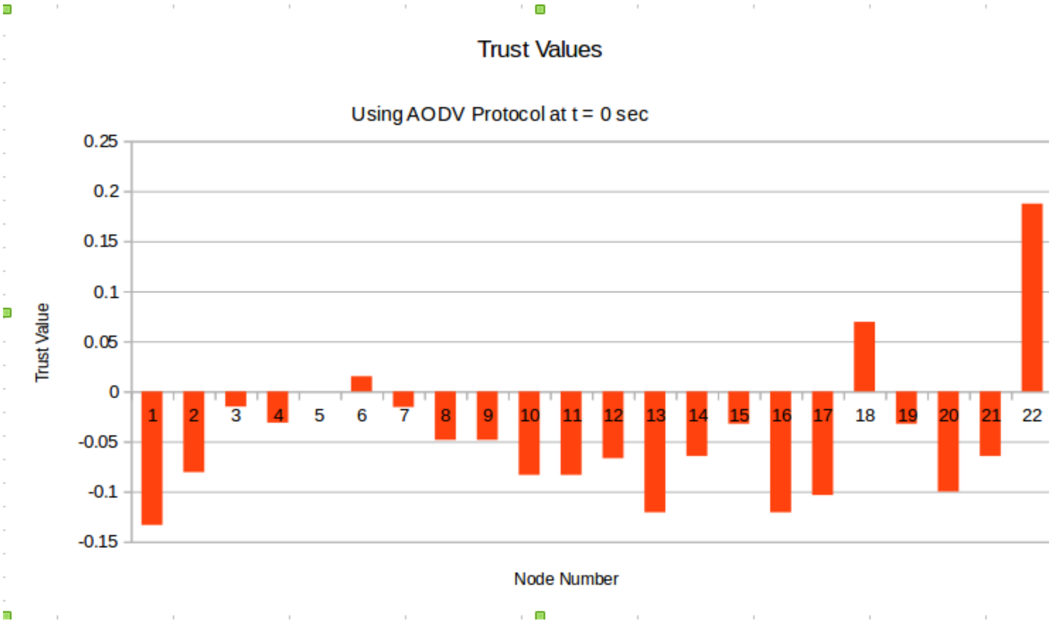


Fig 18 : Trust Value at t = 0 sec

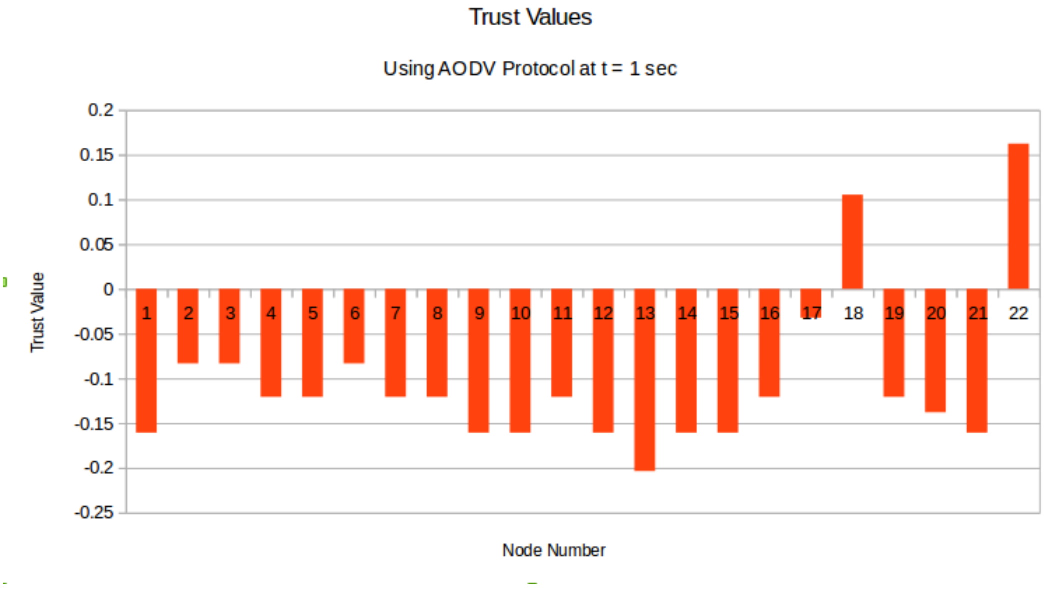


Fig 19: Trust value at t =1 sec

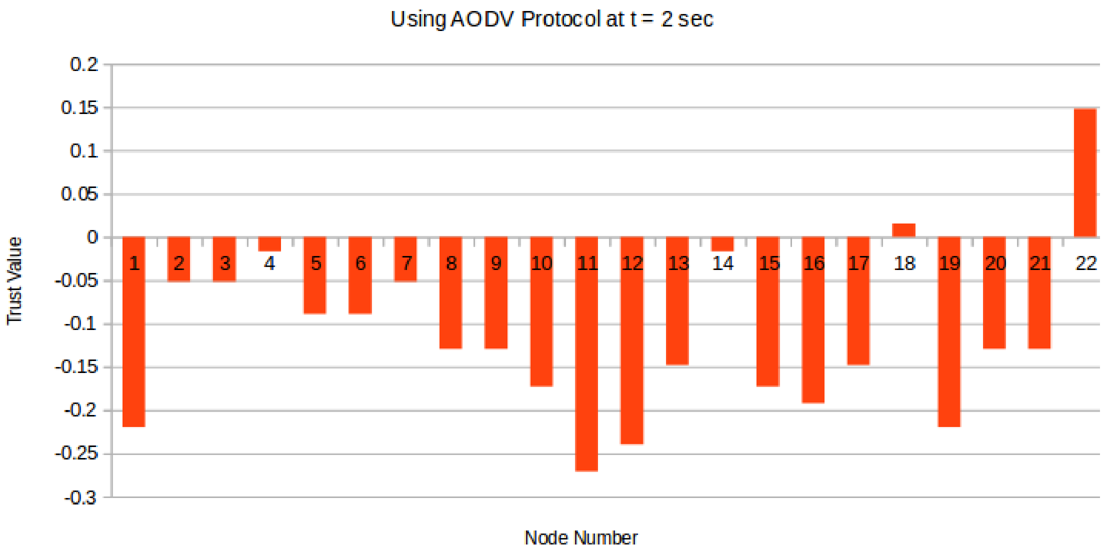


Fig 20: Trust value at t = 2 sec

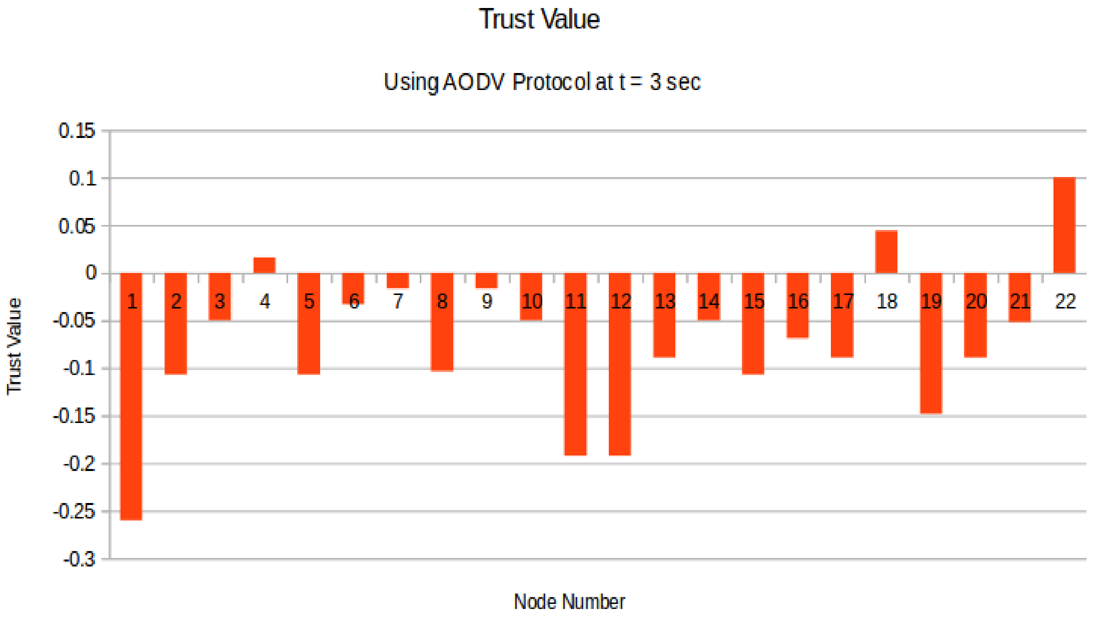


Fig 21: Trust value at t = 3 sec

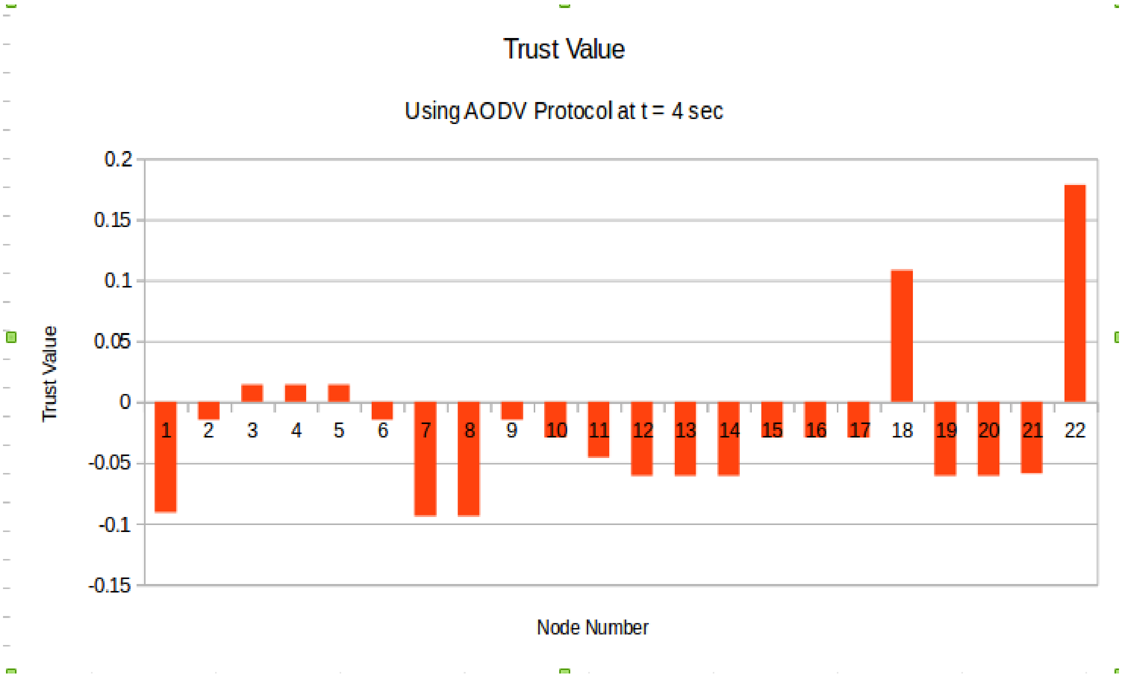


Fig 22: Trust value at t = 4 sec

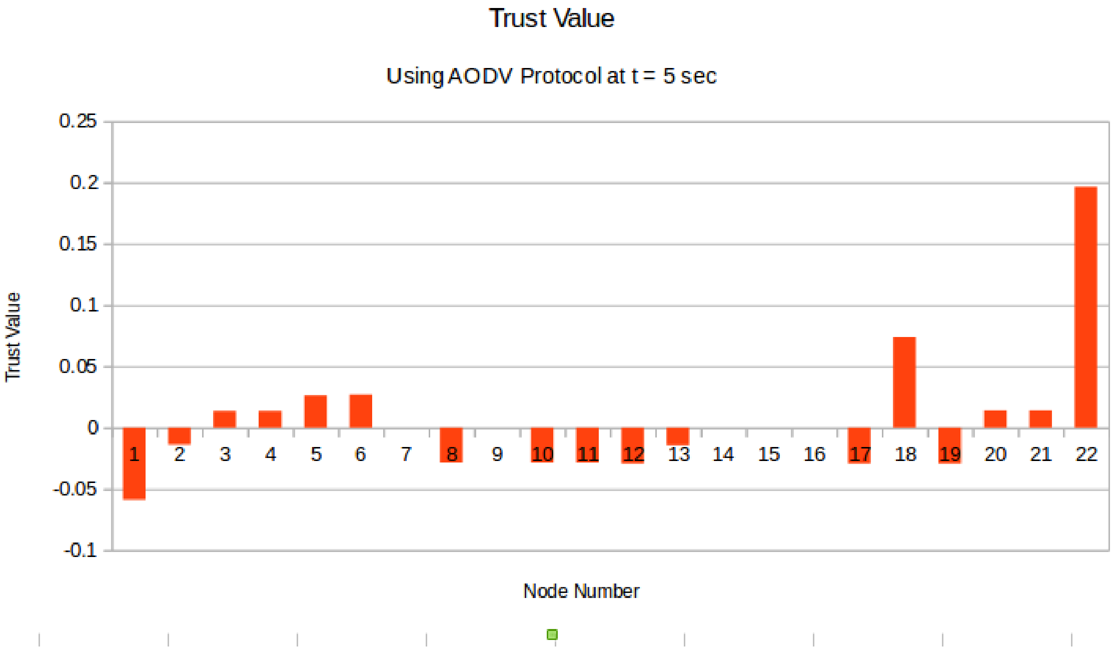


Fig 23: Trust value at t = 5 sec

From the graphs above, the most trust worthy node is node_22 with trust values at t=0, 1, 2, 3, 4, 5 sec are 0.18, 0.1623, 0.1478, 0.1, 0.178, 0.19 respectively and the malicious node is node_1 with trust values at t=0, 1, 2, 3, 4, 5 are -0.13, -0.203, -0.27, -0.26,- 0.09, -0.05 respectively. Now with help of these trust values at different time intervals help us to detect malicious nodes. We can remove the malicious nodes from the network. With the help of the trust value network becomes secure. Using this information we can avoid sending data to malicious nodes. We can handle the confidential data to the trustable node. With the help of the trustable node the packet can be sent to its destination. Our project helps to find out the malicious node and make the network secure.

### 4.7 Calculation of trust factor of each node at time (t) with fixed packet size:

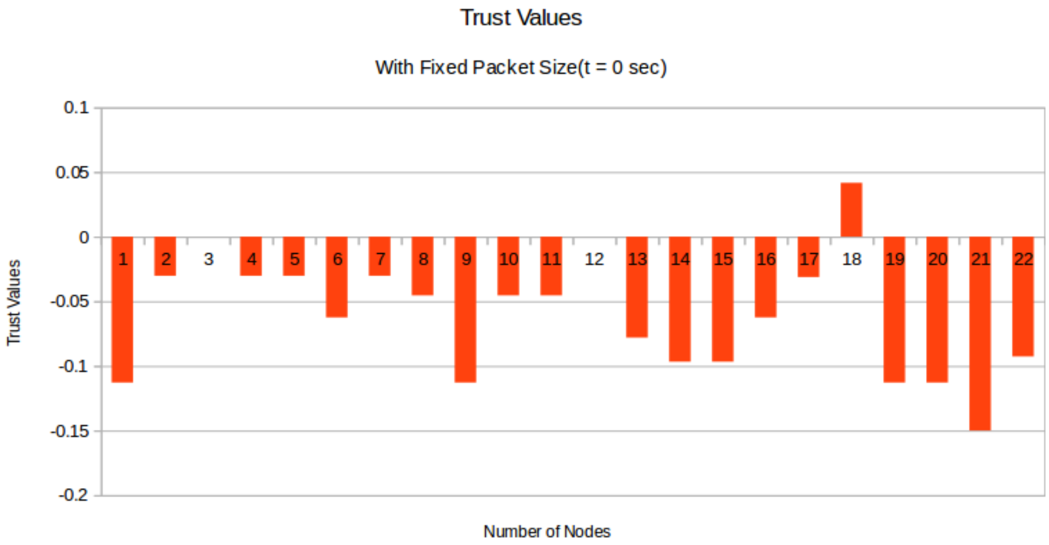


Fig24: Trust Value at t = 0 sec

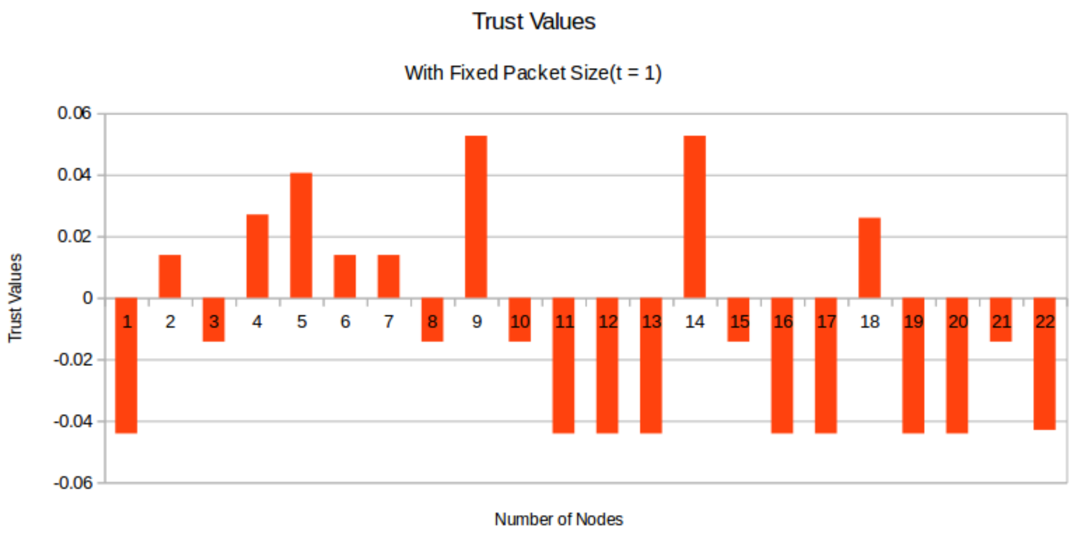


Fig 25: Trust Value at t = 1 sec

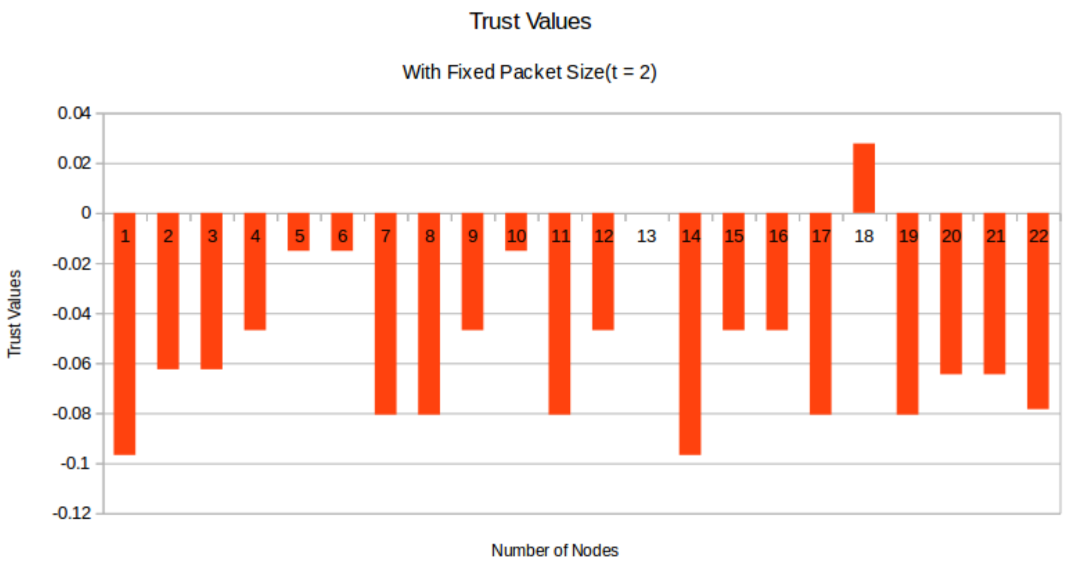


Fig 26: Trust Value at t=2sec

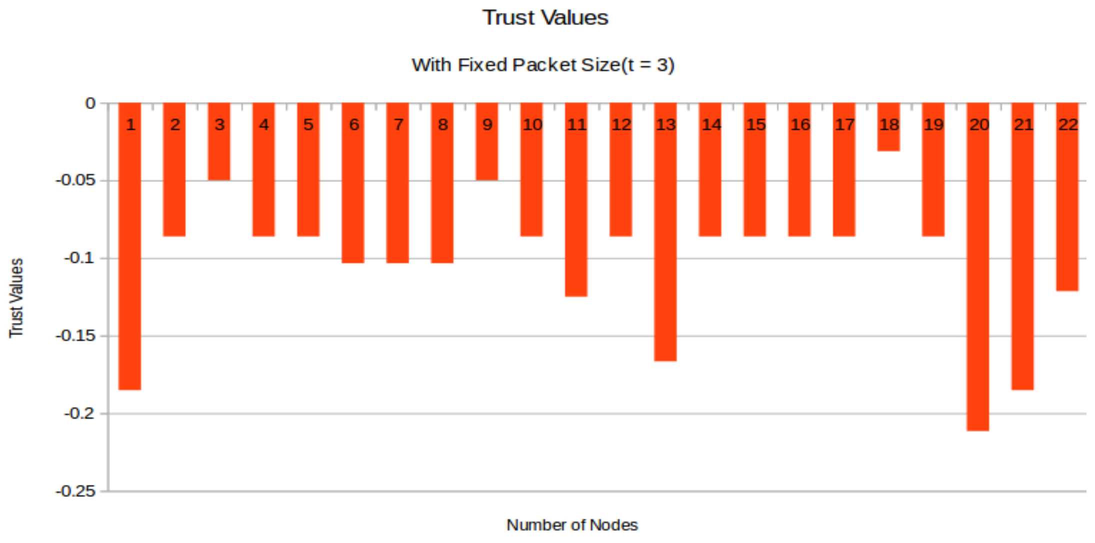


Fig 27: Trust Value at t = 3 sec

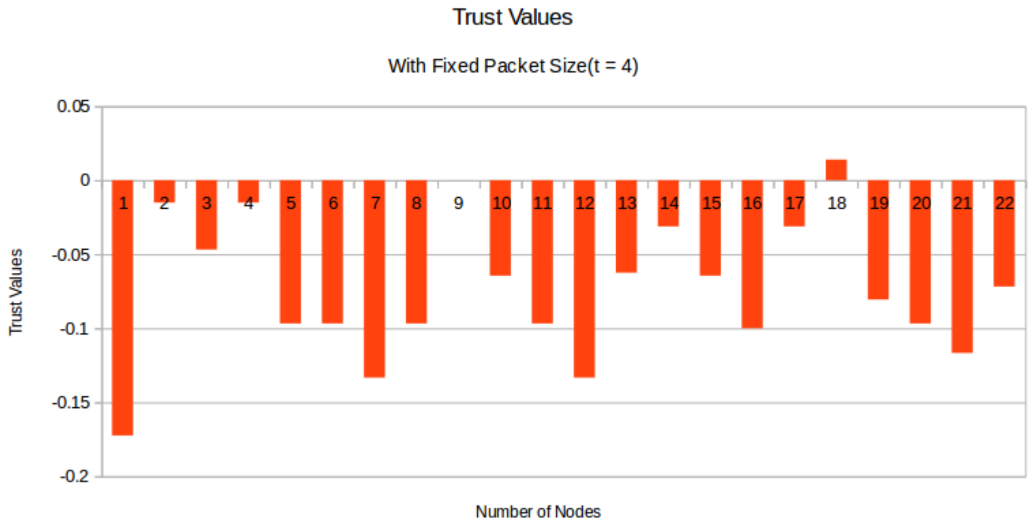


Fig 28: Trust Value at t=4sec

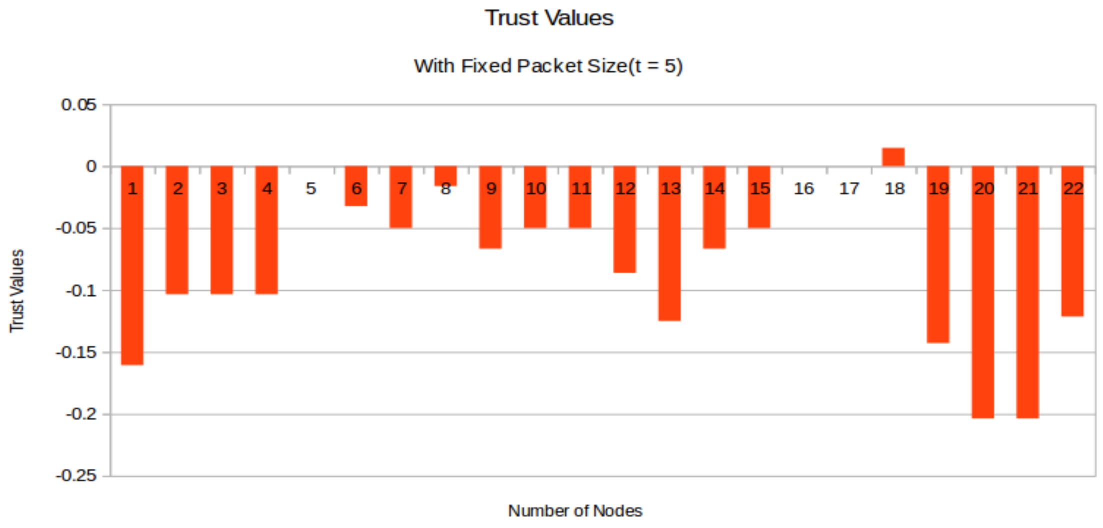


Fig 29: Trust Value at t = 5 sec

The most trustable node when the packet size was fixed to 500 bytes came out to be the 18th node for a particular instance of time. The values are calculated at different time intervals for all the nodes individually at time t = 0, 1, 2, 3, 4, 5 seconds.

The most malicious node was the 1st node amongst all the others with trust values consistently very low at all time intervals.

# CHAPTER 5

# SUMMARY

## 5.1 Conclusion

Ad Hoc network is a network which changes rapidly so, security is a important thing in this network. We studied about how to calculate trust factor so that malicious node can be detected. In our work we studied how to propagate trust and also how to calculate trust on other node's recommendations. Our present work will give us better result in comparison to other researched work. We calculate trust on the basis of two methods first is trust propagation and second is trust recommendation from other nodes. In this we will present a trust metrics in distributed networks and it will be updated on the basis of the calculation of trust.

Security is vital in Ad Hoc Networks. Trust as a concept has a wide variety of adaptations and applications, which causes divergence in trust management terminology. In MANET the participating nodes (devices) works in a collaborative manner, a concrete and attentive trust relationship amongst the neighbors can provide secure network operation. In the proposed solution trust is not calculated for any particular situation. It computes the global trust of the target node depending on its neighbor's recommendations and their own calculated trust levels. A neighbor itself will calculate the percentage based trust and recommend it to the requester. The present work shows that from the existing method, the proposed method is giving results which are almost 25 – 35 % better than the existing one( A Novel Approach for Securing Mobile Ad Hoc Network with an Enhanced Trust Calculation Method Amit Chauhan, Prof. R.P. Mahajan). The node has its own trust value in the table and updating its value while observing the nodes. In this, we present an information theoretic framework for trust evaluation in distributed networks. Based on the proposed study, the level of trustworthiness can be quantitatively determined based on observations and through propagation. In this work, we demonstrate the usage of the proposed models in ad hoc network in malicious node detection and route selection. The simulation results show that the malicious nodes can be detected and the types of their malicious behaviors can be identified. In addition, with the trust recommendations and trust records, the chances of malicious node being on the routes are greatly reduced. We propose a flexible trust model based on the concept of human trust, which provides nodes with a mechanism to evaluate the trust level of its neighbors. The basic idea consists of using previous experiences and recommendations of other neighbors to appraise the trust level of other nodes. In our model, the interactions among nodes are confined to neighbors. Such approach implies lower resource consumption and a lower vulnerability to false recommendations attack. Another important quality is the flexibility due to the possibility of operating in three different modes, depending on the node resource restrictions. At each round of computation, the source node computes opinions for all nodes. This means that information acquired at a single round can be stored and subsequently used for many trust decisions. If there is not enough evidence to determine an opinion, then no opinion is formed. In our work some of the routing issues and change requirement related to trust based routing has been pointed out(Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity-Based Model Pedro B. Velloso, Rafael P. Laufer, Daniel de O. Cunha, Otto Carlos M. B. Duarte, and Guy Pujolle, 2010). These issues are supposed to take good care for developing an efficient, secure and robust routing protocol for wireless ad hoc networks. Taking these issues in to account handling and identification of malicious node can be done easily as well as a model can be developed for calculating trust and analyzing security of the model(A Novel Approach for Securing Mobile Ad Hoc Network with an Enhanced Trust Calculation Method Amit Chauhan1, Prof. R.P. Mahajan2, 2013).

## 5.2 Future Scope

Future work includes defining and implementing a monitoring scheme for a specific application and applying our model to improve the service/application performance, as for instance, an authentication protocol. We also plan to implement more elaborate models for the attackers’ behavior, and for the measures taken against nodes that are being assigned low trust values (i.e., detected to be bad). So, the attackers will be facing a tradeoff between the amount of damage they can inflict, and the possibility of being, for instance, isolated from the rest network. Suitable strategies will be developed for Good as well as Bad nodes. As our future work, we will try to reduce the energy consumption, control overhead and delay of our proposed protocol by applying some optimization techniques (Trust Computations and Trust Dynamics in Mobile Adhoc Networks: A Survey Kannan  Govindan, Member IEEE and Prasant Mohapatra, Fellow IEEE).

There are important issues yet to be addressed. Some of them include:

- Impact of network dynamics on trust: Though, we have given a brief outline about impact of network dynamics on the various trust dynamics, the detailed analysis of the impact has to be addressed. For example, mobility can impact the trust propagations and various other security paradigms. But the clear quantifiable relationship is yet to be determined. Similarly, the relationship between other network dynamics (including link dynamics, network density) and trust and its dynamics are yet to be analyzed.

- Computations of trust in cooperative and noncooperative games: In a self organized distributed network, nodes can give positive or negative recommendations about others either genuinely or maliciously with some self interest. These aspects are analogous to situations in complex systems with game theoretic interactions. The games can be non cooperative where every node plays game independently or cooperative where a set of nodes form sub groups and play game together against the rest of nodes. Non cooperative games are tractable using Nash equilibrium. Trust computation with cooperative game is not well analyzed yet. The earlier attempts are preliminary in nature and these attempts exploit the collaborations in positive way to obtain the trust scores.

- Impact of heterogeneous nodes on trust: Wireless networks could be highly heterogeneous. The heterogeneity could be in terms of the roles of the nodes, their inherent capability and security. Heterogeneity implies that not all nodes or their contents can be treated equally when it comes to trust evaluations. Thus, the same functional descriptions will not be applied to evaluate the trust levels of all nodes and their information. Investigation is needed on incorporating network dynamics and heterogeneity in the trust evaluation functions.

- Security paradigms to enhance trust in the network: The data delivery capabilities and security properties of the network directly impact the level of trust a recipient places on the information received. As an example, it is possible that a piece of information cannot be fully trusted unless its source and the path over which it is received are authenticated. If authentication services are not available one must decide whether to have the untrusted information or none at all. Further research is required to characterize these metrics through modeling efforts and to determine the degree to which security properties influence the network trust.

• Social and context dependent trust: Social relationship and context based trust by establishing social communities among entities has received considerable attention in recent days. However, this is still unexplored area with respect to MANET. The complex dependence between the communications network, the social network, and the application network is not yet explored in MANET. The social communities can also help in validating the trust measurements. Validation of measured trust is another major area of future research. We hope that the near future will bring consolidation around a set of fundamental principles for building trust and its various related issues, and that these will be realized in practical and commercial applications.

### 5.3 Applications Contributions

We intend to integrate an effort-based mechanism like HashCash (Back 2002) into our trust model, to also provide active challenge-response based trust values. For analytical evaluation we are investigating the use of Zero-Knowledge and Game Theory concepts in ad-hoc networks for trust establishment.

We plan to extend our model to other ad-hoc network routing protocols like TORA, AODV and DSDV. We will also look at further issues that have not been addressed in this paper, including trust decay over time, trust acquirement through malicious behavior, malicious colluding nodes, and a security analysis of the proposed model against attacks (Trust Management in Mobile Ad Hoc Networks for Bias Minimization and Application Performance Maximization Ing-Ray Chen, Jia Guo, Fenye Bao).

# REFERENCES

[1] Yan Sun, Wei Yuy, Zhu Hany and K. J. Ray Liuy, "Trust Modeling and Evaluation in Ad Hoc Networks", 2005

[2] www.isi.edu/nsnam/nam/

[3] Dr. Helen and D. Arivazhagan, Applications, Advantages and Challenges of Ad Hoc Networks, 2014

[4] Amandeep Verma and Manpreet Singh Gujral,Trust oriented security framework for ad hoc network

[5] Chung-Wei Hang and Munindar P. Singh, Trust-Based Recommendation Based on Graph Similarity

[6] Peter Stanforth & Vann Hasty, MeshNetworks Inc., Meshing Together: Advantages and Challenges of Deploying Ad Hoc Wireless Networks

[7] Martinus Dipobagio, An Overview on Ad Hoc Networks

[8] Asad Amir Pirzada and Chris McDonald, Establishing Trust In Pure Ad-hoc Networks

[9] Mukesh Kumar Garg, Dr. Ela Kumar, Routing Issues for Trust Based Framework in Mobile Ad Hoc Networks ,2013

[10] Amit Chauhan, Prof. R.P. Mahajan, A Novel Approach for Securing Mobile Ad Hoc Network with an Enhanced Trust Calculation Method

[11] Theodorakopoulos And Baras, On Trust Models And Trust Evaluation Metrics For Ad-Hoc Networks

[12] A Survey by Kannan Govindan, *Member IEEE* and Prasant Mohapatra, *Fellow IEEE* Trust Computations and Trust Dynamics in Mobile Adhoc Networks

[13] Ari Keranen, Opportunistic Network Environment simulator

[14] Jie Zhang, Chen Chen Robin Cohen , A Scalable and Effective Trust-Based Framework for Vehicular Ad-Hoc Networks

[15] Pedro B. Velloso, Rafael P. Laufer, Daniel de O. Cunha, Otto Carlos M. B. Duarte, and Guy Pujolle , Trust Management in Mobile Ad Hoc Networks Using a Scalable Maturity-Based Model

[16] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure ondemand routing protocol for ad hoc networks," in Proceedings of MobiCom 2002, Sep 2002.

[17] Ing-Ray Chen, Jia Guo, Fenye Bao ,Jin-Hee Cho , Trust Management in Mobile Ad Hoc Networks for Bias Minimization and Application Performance Maximization

[18] http://www.cs.binghamton.edu/~kliu/cs580t/ns2pre.pdf

[19]http://www.isi.edu/nsnam/ns/

[20]http://www.cs.virginia.edu/~cs757/slidespdf/cs757-ns2-tutorial1.pdf

[21] http://www.isi.edu/nsnam/ns/tutorial/nsscript4.html

[22]https://www.ietf.org/rfc/rfc3561.txt

[23]https://en.wikipedia.org/wiki/Ad_hoc_On-Demand_Distance_Vector_Routing

[24]https://en.wikipedia.org/wiki/Ad_hoc_On-Demand_Distance_Vector_Routing

[25]https://www.nsnam.org/docs/models/html/dsdv.html

[26]https://en.wikipedia.org/wiki/Python_(programming_language)

[27]https://cloudns2.wordpress.com/awk-files/