# A TUNING BASED MULTICONSTRAINED LINK WEIGHT ASSIGNMENT FOR OPTIMIZED DATA TRANSMISSION

*Dissertation submitted in fulfillment of the requirements for the Degree of*

**MASTER OF TECHNOLOGY**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINERRING**

By


**ASHUTOSH SHARMA**

Enrolment No: 142006

Under the esteemed guidance of

**DR. RAJIV KUMAR**

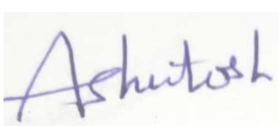
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT, SOLAN
173234, HIMACHAL PRADESH


MAY, 2016

# TABLE OF CONTENTS

# DECLARATION

I hereby declare that the work reported in the M-Tech thesis entitled **"A Tuning Based Multi-Constrained Link Weight Assignment for Optimized Data Transmission"** submitted at **Jaypee University of Information Technology, Waknaghat, India,** is an authentic record of my work carried out under the supervision of **Dr. Rajiv Kumar**. I have not submitted this work elsewhere for any other degree or diploma.



**Ashutosh Sharma**

**(Enrollment No. 142006)**

Department of Electronics and Communication Engineering

Jaypee University of Information and Technology

Waknaghat, Solan, H.P. India- 173234.

Email: sharmaashutosh1326@gmail.com

Date 26/05/2016

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
(Established by H.P. State Legislative vide Act No. 14 of 2002)
P.O. Waknaghat, Teh. Kandaghat, Distt. Solan - 173234 (H.P.) INDIA
Website: www.juit.ac.in
Phone No. (91) 01792-257999 (30 Lines)
Fax: +91-01792-245362

# CERTIFICATE

This is to certify that the work reported in the M-Tech. thesis entitled, "**A Tuning Based Multi-Constrained Link Weight Assignment for Optimized Data Transmission**" submitted by **Ashutosh Sharma** at **Jaypee University of Information and Technology, Waknaghat**, **India,** is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.

**Dr. Rajiv Kumar**
Assistant Professor
Department of Electronics and Communication Engineering
Jaypee University of Information and Technology
Waknaghat, Solan, H.P. India- 173234.
Email: rajiv.kumar@juit.ac.in

Date 26/05/2016

# ACKNOWLDGEMENT

I am deeply indebted to my supervisor **Dr. Rajiv Kumar,** Electronics and Communication Department, for his guiding force behind this project. I want to thank him for introducing me to the field of Communication Network and giving me the opportunity to work under him. In spite of his extremely busy schedules in Department and International Project, he was always available to share with me his deep insights, and extensive experience. His advices have value lasting much beyond this project. I consider it a blessing to be associated with him.

I am also thankful to **Dr. Sunil Bhooshan**, **Professor and Head**, **Dr. Ghanshyam Singh, Mr. Salman Raju Tallui, Dr. Shruti Jain**, **Mrs. Vanita Rana, Miss. Pragya Gupta, Dr. Sunil Dutt Sharma, Mr. Mohd. Wajid** of Electronics and communication Engineering Department, for the motivation. I express my respects to all the faculty members of ECE Department for their generous help in various ways for the completion of this dissertation.

I want also to thankful to my friends and colleagues Ankur Thakur, Jitain Sharma, Sahil Bhusri, and Vivek Thakur for their support and helping hand whenever I need them. All those who are not in list, they are always in my memories.

I am thankful to God for the shower of warm blessings and my parents for their moral support and continuous encouragement while carrying out this study.

Ashutosh

Date: 26/05/2016 Ashutosh Sharma

ABSTRACT

Routing is a technique, which is used to setup a route from source node to destination node. The route chosen either be an optimized route with respect to delay, capacity, reliability, or may be other QoS parameters. This dissertation examines the problem of transmitting data from designated source to destination in acyclic network so that the reliability, delay and capacity play the significant role to the transmission of data. There are number of applications at the user end and to give them quality of service (QoS) it's become necessary to have optimized with application requirement. Path chosen by the network is controlled by the tunable factor to consider the all three factors.

The dissertation will discuss the experimental results, time complexity and performance parameters**.** A time complexity algorithm is presented which guarantees an optimal solution with in a polynomial time.

This dissertation constituent of implementation of the tunable performance extensions and examine their usefulness in routing of traffic. Tunable factor gives the better results with all parameter consideration viz. reliability, delay and capacity by doing the trade-off among all parameters for getting the optimum QoS.

To implement these proposed approach we used NS-2 tool for our simulation using tcl script language and offline tuning is done on the certain set of values.

# LIST OF ABBREVIATIONS

| | |
|---|---|
| CBR | Constant Bit Rate |
| ftp | File Transfer Protocol |
| MRDPT | Most Reliable Data Path Transmission |
| MRTDPT | Most Reliable Tuned Data Path Transmission |
| NS-2 | Network Simulator 2 |
| NAM | Network Animation |
| QoS | Quality of Service |
| Rtproto | Routing Protocol |
| Rtproto DV | Routing Protocol Distance Vector |

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

Today internet has become the bone of the social life which makes us a significant part of the worldwide communications [1]. While internet was born the services were provided with no guarantee issue or called as best affords services [2]. But with new era, things gone advanced and internet applications needed to be fully optimized with their concern references [3]. A major difficulty in the network applications (remote surgery, video teleconferencing, mass data transfer and video conferencing) is to ensuring the optimized services over the internet [4]. A fundamental problem that relies on many important network issues like soft-QoS, internet television, best path selection, optimization of network resources and traffic engineering, is to find the optimized shortest path that satisfy the constraints [5]. For the perfect real time traffic management and QoS, the least delay, maximum capacity and reliability has particular importance. For the traffic engineering to prove the service level satisfaction, the important thing is to find the path that assures the required constrained, while at the same time optimize the network resources [6]. The algorithm for computing the online constrained shortest path can be used in many circumstances depending on the application [7].

## 1.1 Applications of network

Applications of the network are very vast and here some of the following are the applications of network [8];

### 1.1.1 Solar Energy Network Application

Energy plays a important role in human life to run regular life. The application diagram shows the solar energy plan which includes fiber optical cable, wifi-network and router with other network components.

The optical cable connected to the power sub-station plant where network operations take place. Main operations are help at station are control of power distribution and load balancing.

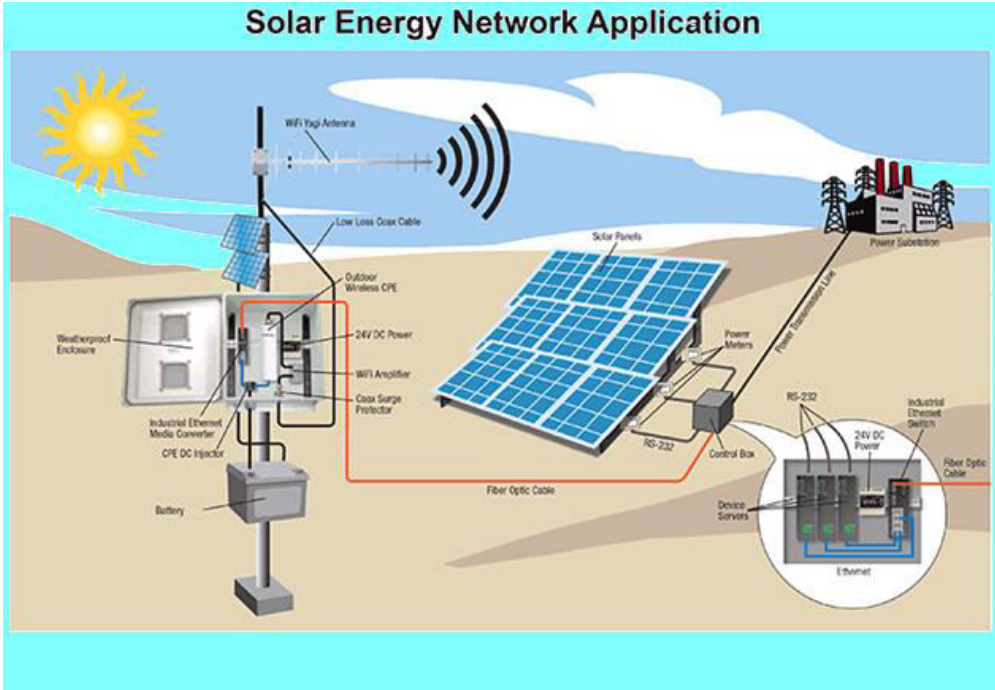



**Figure 1.1:** Solar Energy Network

### 1.1.2 Process Automation Network Application

In the large factories or plants a number of machines that are work together and that is difficult to observe that individual.

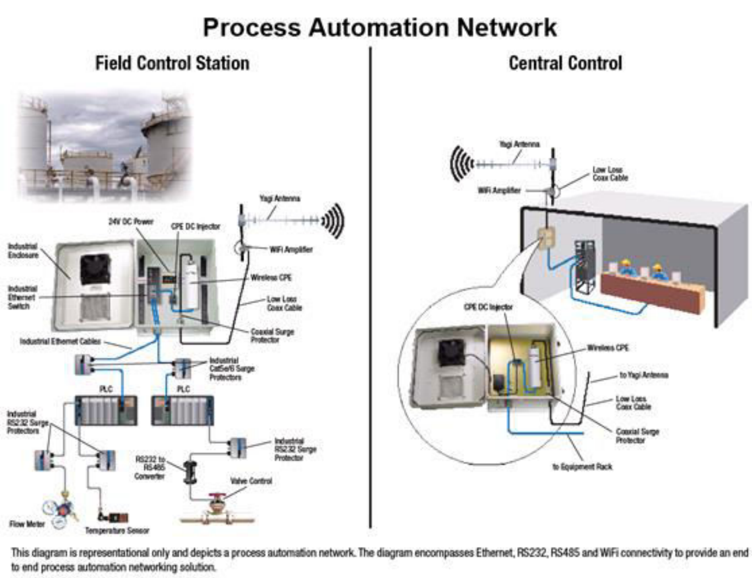



**Figure 1.2:** Process Automation Network

To observe them whether they are working ok or not large empowerment requires. To reduce that a centralized automated network is designed. All equipment is connected with centralized system via Ethernet and Wi-Fi network.

### 1.1.3 Wind Farm Network Application

A major power sources for the social human life is wind power. The network is used to monitor and control the wind turbine by network administrator at far apart location connected through cables or may be wireless link.

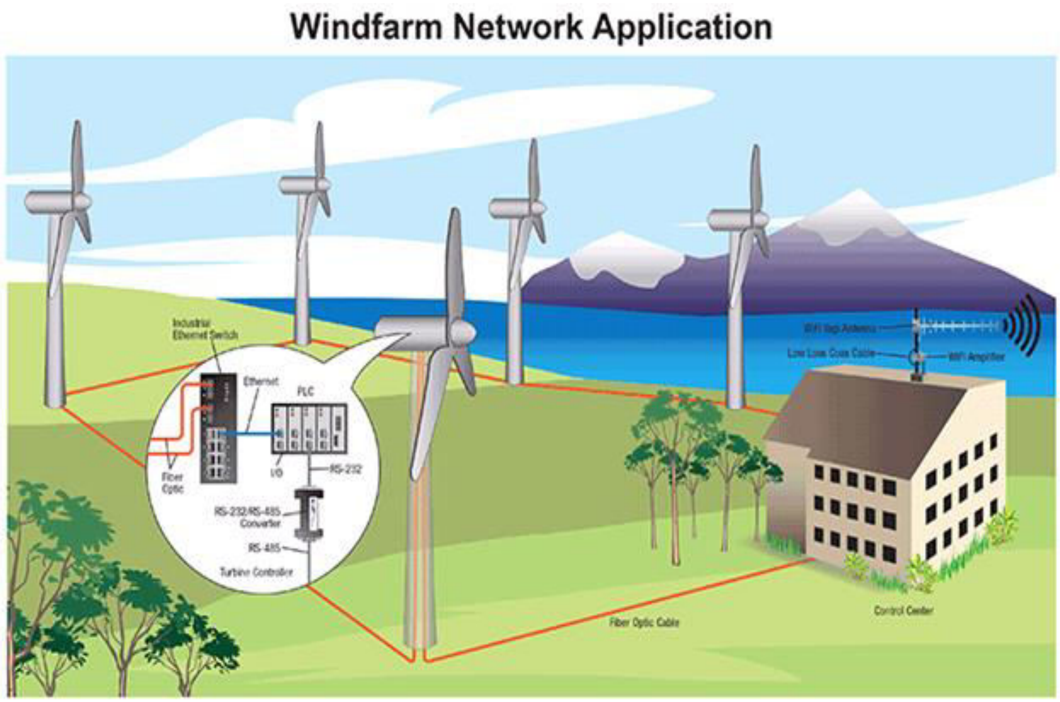



**Figure 1.3:** Wind farm Network Application

### 1.1.4 Telephonic Surgery

The advancement in the network applications leads to use the critical services like telephonic surgery or telemedicine. In the field of medical sciences now a day's surgeries are done over network which requires perfect link between ends to end. Telephonic surgeries are becoming so much popular with time. The doctors from world do the major and critical operations using this technique.

**Figure 1.4: Telephonic Surgeries**

### 1.1.5 Waste Water Treatment Plant

The wastewater treatment plant network is the best example of utilizing the network applications. The settling pools used in the waste water treatment plant are monitored from a center control room through copper cable and Ethernet cables.

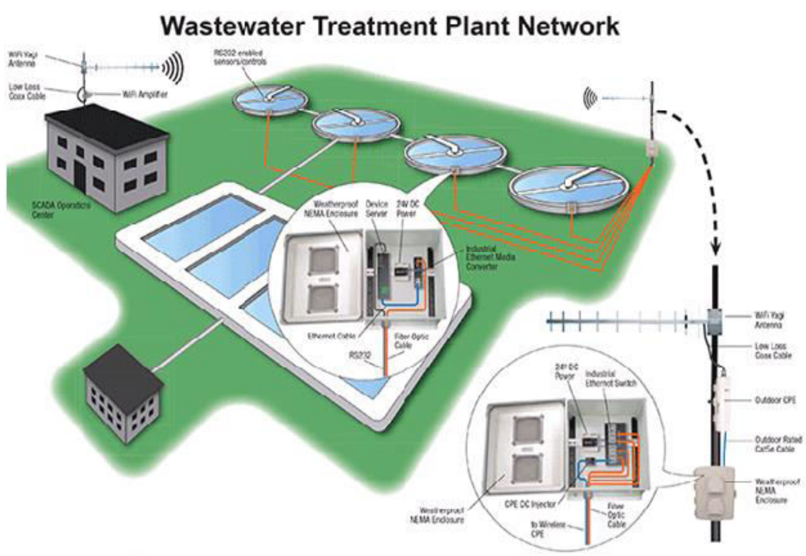



**Figure 1.5:** Waste Water Treatment Plant

### 1.1.6 Online solutions

Number of companies starts giving the online solution for their vendors. These services are based on the network which is connected either through Ethernet wires, optical wires, microwave links or wireless links.


**Figure 1.6:** Online Solutions

This dissertation investigates the three constraints as reliability, capacity and delay in optimized tunned routing. The reliability defined here is the packet delivery ratio. However the reliability constraint is not a key factor on the optimization of network resources as compare to capacity and delay [9]. There are the numerous papers presented on the optimized routing but very few of them have taken the multiple constraints. A major hurdle in multi-constrained routing is the time complexity and/or space complexity because finding the path based on two or more constrained is NP-complete problem [10]. One of the most discussed topics in the internet research community is the tunned and automatic configuration of the network with constraints of reliability, capacity and delay [11].

Delay, reliability and capacity are needed to be satisfied in different ways means to say that delay is time constraint, reliability depends on the path diversity and capacity is the constrained based on bandwidth [12]. The multi-constraint multipath proposal is based on two categories. One category gives the computed various possible paths from source edge to destination and other category will combines the resources to the computed tunned path with requited constraint [13].

The objective of the algorithm is that the path is chosen for the routing should be specific to application requirement so that performance parameters will not undergo any drastic effects.

## 1.2 Related Work

#### 1.2.1 Title: A Link Weight Assignment Algorithm for Traffic-Engineered Networks

Banerjee and Sidhu [14] proposed two algorithms first one takes care of bandwidth constraint named as (TE-B) and second one considers both bandwidth as well as delay constraint (TE-DB). Proposed algorithm introduces two objectives in the traffic engineering as well as load balancing with the optimized path for the data transmission. Each objective is achieved by selecting the shortest path first according to the appropriate link cost. Since traffic or data can be random in nature so problem is NP-hard.

#### 1.2.2 Title: Dynamic Routing in QoS-Aware Traffic Engineered Networks

S Avallone [15] gives the QoS driven path in which the quality of service is not the issue of service providers so to make paths feasible to user it gives the new algorithm which is called as SAMCRA. The proposed algorithm is compared with SAMCRA-B which has a constraint of bandwidth considered with link. A minimum cost driven path is chosen when there are feasible path are present. Otherwise the paths with low cost are taken with reference to infeasible paths. The path chosen is merely a tightened path for the transmission.

#### 1.2.3 Title: On the complexity of QoS routing

Miegheem and Kuipers [16] used the Dijkstra's algorithm to reduce the computation time of path search at the cost of relaxing exactness of solution.

#### 1.1.4 Title: The most reliable data-path transmission

Spyros Tragoudas [17] used the MRDPT formula for the cost calculations and further used the Dijkstra's algorithm for the computation of least cost path to result in the optimized values based on the most reliable path. The dependency on the delay and capacity is not that much weighted as reliability.

## 1.3 Motivation

The motivation behind this dissertation is to examine the problem of selecting a path from designated source to designated destination, so that path tuned for the routing should be most reliable path with consideration of reliability, delay and capacity. If path selected has much longer delay than required, that path is not feasible. Same as if path chosen for end to end transmission has not much capacity for transmission of data then that path is not reliable. Many schemes have been proposed to improve reliability, capacity and delay of the transmission path. An effort to reduce the delay and enhance the capacity and reliability of the chosen data transmission path leads to give the most reliable path for the various real time applications and hence we proposed a user-tunable mechanism for selecting router based on factors viz. reliability, delay and capacity known as MRTDPT [17].

Number of papers has been proposed to target reliable path for data transmission problems. Spyros Tragoudas [17] gives the most reliable data path transmission formulating problem based on reliability, capacity and delay constraints. The MRDPT problems consider a simplified problem formulation where each edge considers the reliability, capacity and delay in such a fashion so that delay and capacity are the powers of reliability.

Cost function given by the MRDPT is:

$$\alpha(u,v) = -\ln r(uv)^{\left[d(u,v)+\frac{\sigma}{c(u,v)}\right]} \qquad (1.1)$$

This MRDPT problem leads us to give multi-constrained data transmission path. But as the network application goes vast and wide the requirement of constrains are become different at every miracle of seconds. So to make it flexible we further step up for tuned multi-constrained data transmission path.

### 1.4 Problem Statement

The problem in MRDPT is that it takes account in to cost which is the sum of individual but with respect to reliability, it is the product of individual one. But the path chosen is not so optimal so that tuning factor is introduced with gives the best optimal path for routing by doing the trade-off between the major factors [18]. The MRTDPT formulation does not only depend on the reliability, capacity and delay but also on the data units to be transmitted through the network [19]

### 1.5 Organization of Dissertation

Complete dissertation is organized as below. Chapter 2 discuss about the routing protocols with their comparative study of performance parameters. Chapter 3 enlightens the proposed work that we have implemented in this dissertation with detailed explanations. Chapter 4 explains the setup environment and results. In chapter 5 we concluded the dissertation and future direction of this work.

# CHAPTER 2

# ROUTING PROTOCOLS AND PERFORMANCE METRICS

## INTRODUCTION

A realistic comparison analysis of the static and dynamic routing protocols has been made. Parameters of performance selected for this analysis are average delay, average throughput and average packet loss in different mesh network. Both the routing protocols are used by the network architectures and designers in practice.

A static routing protocol does not check the connection once it has been established i.e. during transmission of data transfer whereas dynamic routing protocol periodically checks the connection and also update the path accordingly. Analysis conducted in this work presents that the dynamic routing is better than static routing in term of average delay and average packet delivery ratio while static routing is better than dynamic routing with respect to average throughput. All the simulations have been done with the help of Network Simulator Tool NS2.

In the present scenario, computer communication network is a necessary entity that is widely touched all over the world [20]. A network may be wired or wireless according to the network planners. Both have certain advantages and limitations [21]. Communication networks primarily have components such as switches, hubs, routers and bridges as its nodes. Routers used to communicate with different network nodes within the network with the help of routing protocols. Two such specified nodes are ‘s’ source and ‘t’ destination nodes. There may exist various end-to-end connections between ‘s’ and ‘t’ nodes corresponding to the path. Here we are considering static and dynamic routing protocols for making end-to-end connections. In the static routing the strategy of routing is the default route computation with the help of some parameters like cost, bandwidth and delay [23].

**Figure 2.1:** Routing Mechanism for Different Nodes [22]

The routing protocol is computed once in the simulation. In dynamic routing the protocol is runs over the algorithm distributed bellman ford or (distance vector) routing. In text various techniques for routing have been mentioned such as Optimized Link State Routing Protocol (OLSR), Destination-Sequenced Distance Vector routing (DSDV), Open Shortest Path First (OSPF) [24, 25] etc.

In this dissertation a Brief Overview of unicast routing protocols and its types has been given with their advantages and limitations.

## 2.1 Unicast routing

In computer networking, unicast routing [26] is used to send a message from source to destination in the single network which is identified by the unique address. Unicast routing is used where traffic is forwarded from source to destination with in the network with unique address [27].

Types of unicast routing protocols

**2.1.1 Static unicast routing protocols:** In this type Dijkstra All-pairs Shortest Path First (SPF) is used and this is calculated once before start of simulation.

**2.1.2 Session unicast routing protocols:** this type is also same as static and uses Dijkstra algorithm but difference is that it will do recalculation when there is any change in topology. It is central routing protocol.

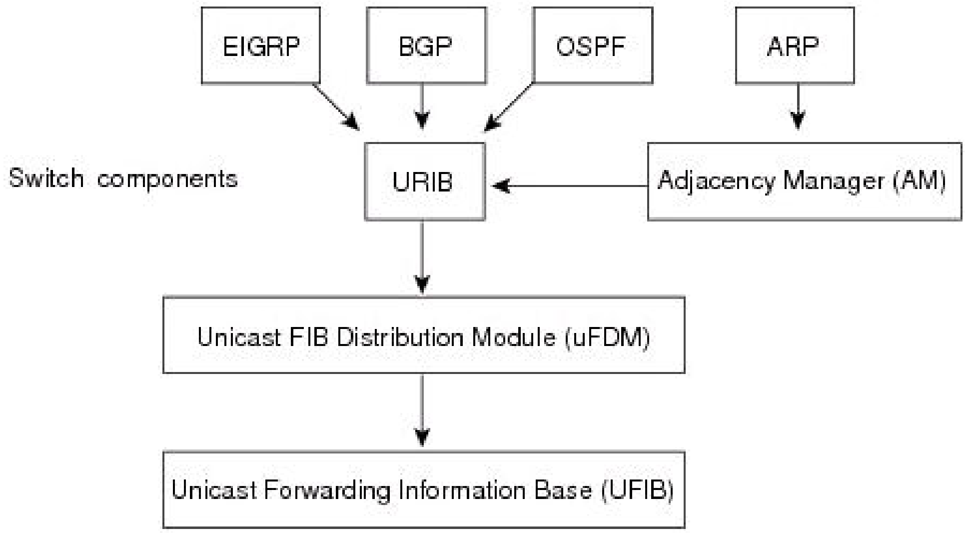



**Figure 2.2:** Flow Chart of Working of Unicast Routing Protocol [28]

**2.1.3 Dynamic (DV) unicast routing protocols:** in this type of routing an agent is created per node and based on distance vector (Distributed Bellman-Ford) algorithm. This type is based on costs of link and supports for multiple paths to the same destination in the network.

**2.1.4 Manual unicast routing protocols:** in this type no protocol is used it is just equivalent to route command in the network we have to give path manual.

## 2.2 Advantages and limitation of static routing protocol

Static routing protocol takes very less memory or CPU overhead so due to this, it is very efficient in bandwidth allocation. In static routing congestion is not a problem because routes are not updated periodically and calculates the path only once in the starting. In static routing administrator has full control over network

However, as a limitation network topology is adjusted manually if any change or fault occurs in the network

## 2.3 Advantages and limitation of dynamic routing protocol

Dynamic routing protocol is simpler to configure on larger networks. If any link goes down it has been chosen a different (or better) route between sources to destination. It is very adaptive in load balancing between multiple links of a network.

However, as a limitation the choice of best path is not in the hands of network administrator. Also in dynamic routing protocol no. of updates are shared between routers, due to this it is bandwidth consuming and additional load on Center Processing Unit/Read Only Memory (CPU/RAM).

## 2.4 Dijkstra all pair shortest path first

In Static routing protocol (OSPF), Dijkstra's algorithm has been used for choosing the shortest path. Dijkstra's algorithm [29] solves the single-source shortest path first problem when all edges have non-negative weights. In this algorithm [30] starts at the source vertex, S, and it expand a tree, T, that covers all vertices reachable from S. Vertices are added to T in Order of distance which means that first S, then the closest vertex to S, then next closest, and so on. Implementation given bellows shows that graph G is represented by adjacency lists.

Initialization of all nodes set with distance infinite except source node. Source node has to be set with 0 and setting it as active. Mark the distance of source node as permanent and all other distances are temporarily. Calculate temporary distance of all neighbor nodes of the active node by summing up its distance with the weights of the edges. If calculated distance of a node is smaller than current distance, update the distance and set the current node as antecessor. This step is the central idea of algorithm. After updating distances, set node with minimal temporary distance as active and mark it as permanent. These steps has been repeated until no any single node left with permanent distance, which neighbors still have temporary distances.

### 2.4.1 Pseudo code

```
DIJKSTRA (G, w, S)
    • Initialize single-source (G, S)
    • S-{}  // this will contains vertices of final shortest-path weights
      from S
    • Priority queue Q is initialized i.e. Q-V [G]
    • If priority queue Q is not empty do
    • u- extract min (Q)  //gives new vertex
    • S- S E{u}    //Put each vertex v as selected adjacent to u
    • For each vertex v in Adj[u] do
    • Select the (u, v, w)
```

### 2.4.2 Advantages and limitations:

Once this algorithm is carried out it will give least cost path for all static nodes but the limitation for this algorithm is that it will not calculate negative weight arcs. If any negative weight arc occurs then it has given acyclic graph and most often cannot give shortest path.

## 2.5 Distributed Bellman-Ford

In dynamic routing protocol, the Bellman-Ford algorithm [31] finds the shortest paths, from a given source node to all other nodes in network. The general idea of using this algorithm is that it finds the shortest single arc path and then shortest path of at most two arcs [32, 33]. This algorithm calculated the shortest path as given in the steps:

### 2.5.1 Pseudo code

- Graph and source vertex.
- Find shortest to all vertices from source. If there is a negative weight cycle, then shortest distances are not calculated, negative weight cycle is reported.
- First step is initialized distances from source to all vertices as infinite and source at zero.
- Create an array of distance dist[] of size |v| with values as infinite except dist[source vertex]
- This step calculates shortest distances and do following |V|-1 times where |V| is the number of vertices in given graph
- Repeat this for each edge u-v. If dist[v]>dist[u] +weight of edge uv, then update dist[v] else no update.

### 2.5.2 Advantages and limitations:

Bellman-ford algorithm maximizes the performance of system and also it updates the paths after periodic time intervals. So if any changes are there in network it will compute path again. The limitation of this algorithm in RIP is that it does not take weights into consideration.

## 2.6 Network Performance Evaluation

Performance is the major part of network management and network administrator [34] always set it as good as possible for a given cost. Three parameters evaluate the performance depending on analytical designing, simulation and analysis. Simulation is being most important due to its accuracy, time, cost and less assumption.

There are no of qualitative and quantitative performance parameters [35] that can be used to compare routing protocols. This dissertationhas been considered the following parameters to evaluate performance of routing protocols.

**2.6.1 Average Throughput:** This is the rate (bits/time unit) at which bits transferred between sources to sink for a longer time period [36]. Instantaneous throughput is the rate at a given point in time

$$\text{Average Throughput.} = \frac{\text{Total no of bytes received}}{\text{Total time of transmission}}$$

**2.6.2 Average Delay:** This parameter represents average delay and indicates how much time it has been taken by the packet to travel from source to destination and measured in seconds [37].

$$\text{Average Delay} = \frac{\sum(\text{Arrive time - send time})}{\text{Number of connection}}$$

**2.6.3 Average Packet Delivery Ratio:** Average packet delivery ratio is calculated by dividing the number of packets received by sink through the number of packet originated by the source [38]. This average specifies that rate of packet loss in network which leads to limit the throughput of network.

$$\text{Average Packet Delivery Ratio} = \frac{\text{Number of packet received}}{\text{Number of packet send}}$$

## 2.7 Outline of the work

In this section, simulation has been done using network simulator tool. NS-2 is an open source simulation tool [39, 40] and written in C++ and OTcl. In beginning it is difficult for first time user, because very few user-supportive manual. One can found it easy when he gets into it. NS-2 is chosen as a simulator tool among other simulation tools because it is used for designing new protocols, comparison of different routing protocols and various performance parameters. It is also freely available for all platforms like Mac OS, Linux, Windows and large number of people use for development and research.

Here simulation has been done using 7 nodes and 15nodes mesh topology for performance comparison of both static and dynamic routing protocols. The simulation setup has been given in the table.

**Table 2.1:** Simulation Setup

| Parameter | Value |
|---|---|
| Protocol | Rtproto DV, Rtproto Static |
| Traffic source | Constant bit rate (CBR) |
| Packet size | 500 bytes |
| No. of nodes | 7,15 |
| Application | UDP |
| Simulation time (sec) | 5 |

The duplex link has been used for connecting the nodes with drop tail queue. For static routing rtproto static agent and for dynamic routing rtproto DV agent has been used. UDP protocol is used for transmission of data between nodes. CBR application generate the traffic of 80 Kbps attached at node 0 with source and sink is attached at node 6 in 7 node mesh topology and at node 14 in 15 node mesh topology. Traffic using CBR has been started at .5 second and stopped 4.5 second. Using trace file generated by the simulation, the performance analysis has been observed.

## 2.8 Performance analysis

In this section, the simulation of the mesh networks with different nodes, the observation of performance Parameters are shown below like average throughput, average packet loss and average delay. These parameters are explained one by one using help of charts.

**2.8.1 Instantaneous throughput:** In fig. 1instantaneous throughput for static routing protocol and fig. 2 for dynamic routing protocol has been shown that dynamic routing is good because packets sent through constant bit rate (CBR) using UDP agent are started at 0.5 sec, so in dynamic routing is done fast and at .5 it gives the instantaneous throughput of 14.6 for seven nodes and 82.86 for 15 nodes.

**Table 2.2:** Instantaneous throughput of static and dynamic routing protocol with different number of nodes

| Time (sec) | Mesh topology with 7 nodes | | Mesh topology with 15 nodes | |
|---|---|---|---|---|
| | Static | Dynamic | Static | Dynamic |
| 0.5104 | -------- | 14.8639 | --------- | 82.8646 |
| 1.0104 | 2310.15 | 2352 | 5155.88 | 5264 |
| 1.5104 | 2400 | 2400 | 5600 | 5600 |
| 2.0104 | 2400 | 2400.45 | 5600 | 5602.4 |
| 2.5104 | 2400 | 2402.46 | 5600 | 5606.72 |
| 3.0104 | 2400 | 2400 | 5600 | 5601.44 |
| 3.5104 | 2400 | 2400 | 5600 | 5600 |
| 4.0108 | 2400 | 2406.86 | 5600 | 5605.44 |
| 4.5108 | 2400 | 2402.13 | 5600 | 5608.64 |

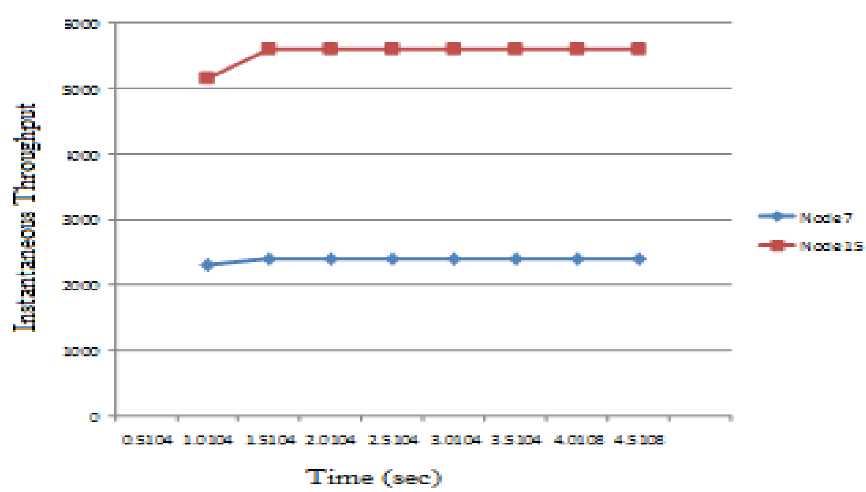



**Figure 2.2:** Instantaneous throughput for static routing protocol (bit/sec)

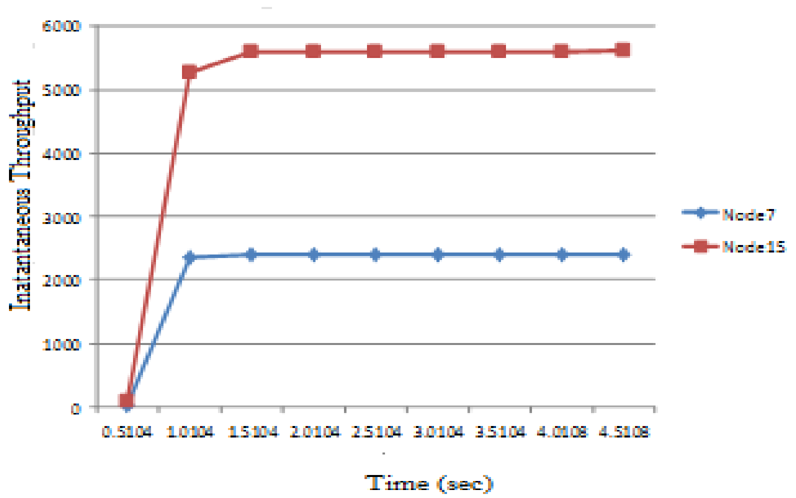



**Figure 2.3:** Instantaneous throughput for dynamic routing protocol (bit/sec)

**2.8.2 Average throughput:** In our observation from fig. 3 we see that best average throughput is shown by static routing protocol. It can easily observe that dynamic routing protocol has low throughput. This lacking in the performance indicated that dynamic routing is not good with network

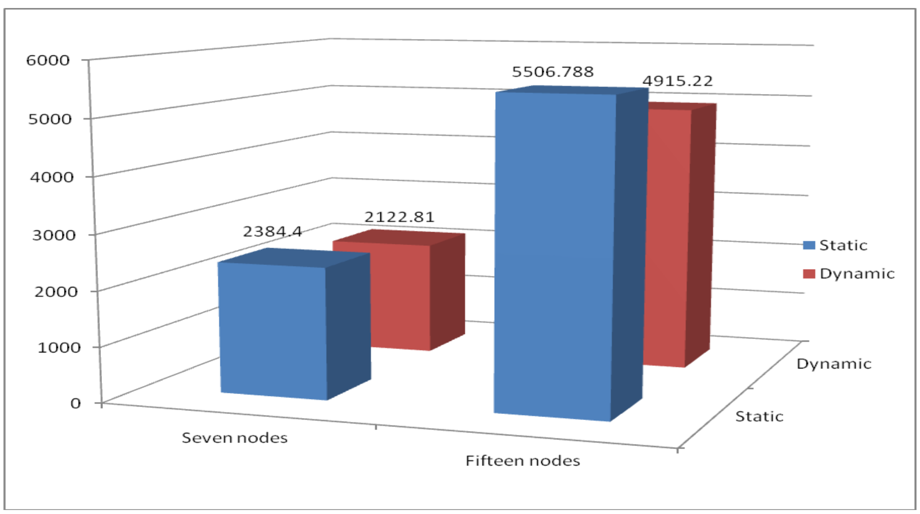



**Figure 2.4:** Results of average throughput (bit/sec)

**2.8.3 Average delay:** Average delay in fig. 4 shows that delay is worst in static routing protocol in comparison with dynamic routing. Due to fast conversion of routing path in dynamic routing lower the delay in network. With increasing the number of nodes it can easily examine that average delay has avalanche of delay in static routing protocol with comparison of dynamic routing protocol.

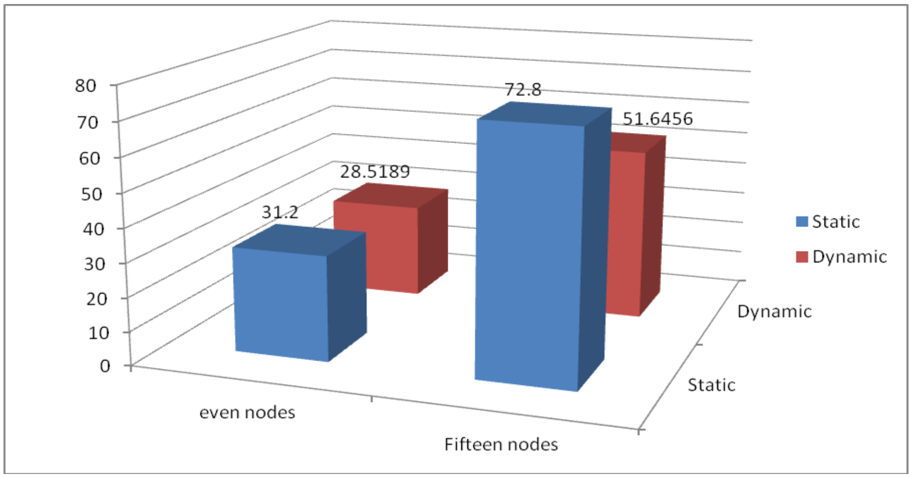



**Figure 2.5:** Results of Average delay (msec)

**2.8.4 Average packet delivery ratio:** Fig. 5 depicts the average packet delivery ratio, the average packet delivery ratio of dynamic routing protocol is better than static routing protocol.

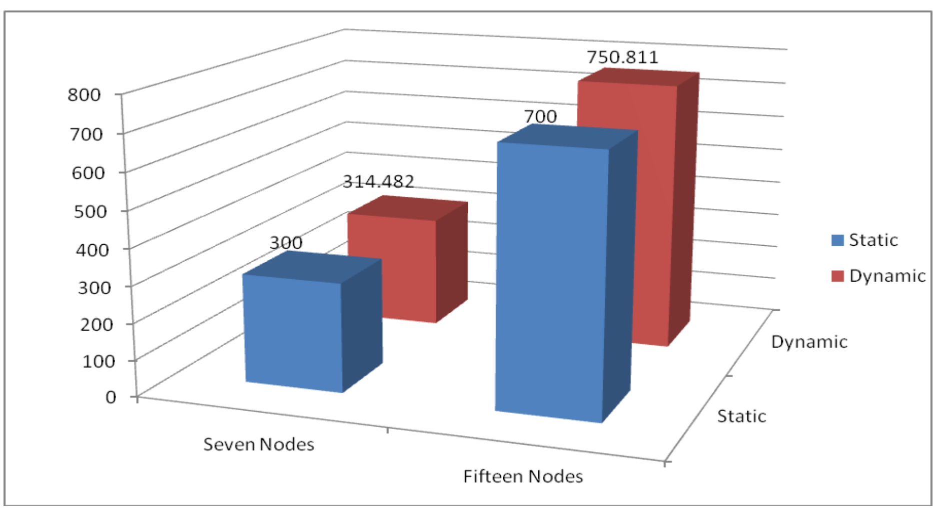



**Figure 2.6:** Results of average packet delivery ratio

We can explain the good performance of dynamic routing protocol with remark that when it finds the best path in between routing, it takes that route which utilizes the bandwidth in best way.

## 2.9 Conclusion

This report makes the realistic comparison of static and dynamic routing protocols using NS2 simulator. By observing the results we examine that the dynamic routing is better than static routing in terms of average delay and average packet delivery ratio. Static routing also has certain merits over dynamic routing particularly in terms of average throughput.

In next chapter we will use the best protocol for our approach according to these results to getting the optimized trade-off between parameters.

# CHAPTER 3

# A TUNNED LINK WEIGHT ASSIGNMENT ALGORITHM

Although wired network is bulky, rigid and immobile, it can be capable of accomplish various applications, such as emergency alarm, telephony surgery, telemedicine, habitat monitoring and many more [41]. Depending on the requirement of application one can set parameter as their usefulness. Like if anyone wants to do skyp, needs to be less delay and same as in telephony surgery and if someone wants to transfer huge data requires high capacity. So optimal routing plays an important role in selecting a most reliable path for data transmission [42].

Consider a directed acyclic network with the pre-computed edges with reliability, delay and capacity values [43]. This dissertationexamines the problem of selecting a path from designated source to designated destination, so that path tuned for the routing should be most reliable path with consideration of reliability, delay and capacity. Delay, reliability and capacity need to be satisfied in different manners. If path selected has much longer delay than required, that path is not feasible. Same as if path chosen for end to end transmission has not much capacity for transmission of data then that path is not reliable. Many schemes have been proposed to improve reliability, capacity and delay of the transmission path. An effort to reduce the delay and enhance the capacity and reliability of the chosen data transmission path leads to give the most reliable path for the various real time applications and hence we proposed a user-tunable mechanism for selecting router based on factors viz. reliability, delay and capacity known as MRTDPT.

Similar reliability problem on path have been studied. The MRDPT problems consider a simplified problem formulation where each edge considers the reliability, capacity and delay in such a fashion so that delay and capacity are the powers of reliability [17]. The given idea does depend mainly on reliability and poorly on delay and capacity. The MRDPT problem was solved with the well-studied shortest path first algorithm in networks. To make the path chosen for transmission totally dependable on reliability, capacity and delay we have proposed MRTDPT which is more realistic than MRDPT. MRTDPT can be solved optimally by modifying well studied shortest path first algorithm [44, 45].

The problem in MRDPT occurs here is that it takes account in cost which is the sum of individual but with respect to reliability, it is the product of individual one. But the path chosen is not so optimal so that tuning factor is introduced with gives the best optimal path for routing. The MRTDPT formulation does not only depend on the reliability, capacity and delay but also on the data units to be transmitted through the network [46].

The MRTDPT can be used in number of applications with different size of data units routed from source to sink. Due to the pre-computed paths, it guarantees the most reliable path [47].

## 3.1 Preliminaries

Number of papers has been proposed to target reliable path for data transmission problems. Spyros Tragoudas [17] gives the most reliable data path transmission formulating problem based on reliability, capacity and delay constraints. As given that σ units of data are to be transmitted from source s to destination t with the routing cost consideration of all three factors reliability, capacity and delay to give MRTDPT though (u,v), the reliability is shown as:

$$E(u,v) = -\ln r(uv)e^{-\left[d(u,v)+\frac{\sigma}{c(u,v)}\right]} \qquad (3.1)$$

**The above equation has been proposed on the equation given below as MRDPT problem.**

$$\alpha(u,v) = -\ln r(uv)^{\left[d(u,v)+\frac{\sigma}{c(u,v)}\right]} \qquad (3.2)$$

Equation 2 gives most reliable path from source to destination but that depends more on reliability and less on capacity and delay.

## 3.2 Example 1

Let's consider a topology of 8 nodes with traffic size σ=5 units with given set of values of reliability (r), capacity (c) and delay (d). When data is transmitted from source to destination the path chosen is 0-2-7-6-1-5 which has reliability 0.31 delay 29 sec and capacity 0.80.

Here in this problem more concern is more on reliable path chosen for transmission and less on delay and capacity.

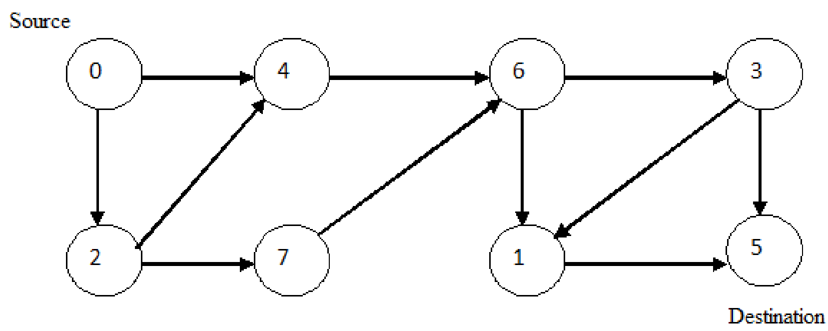



**Figure 3.1:** Topology G (s, t)

**Table 3.1:** Pre-computed set of values for the links (u, v) for the topology G(s, t)

| Edge(u, v) | Edge Reliability r(u,v) | Edge Delay d(u,v) | Edge Capacity C(u,v) |
|---|---|---|---|
| (0,2) | 0.7 | 7 | 0.80 |
| (0,4) | 0.7 | 5 | 0.50 |
| (1,5) | 0.7 | 4 | 0.95 |
| (2,4) | 0.8 | 5 | 1.00 |
| (2,7) | 0.9 | 7 | 1.50 |
| (3,1) | 0.8 | 3 | 0.70 |
| (3,5) | 0.7 | 6 | 0.76 |
| (4,6) | 0.6 | 6 | 0.80 |
| (6,1) | 0.9 | 5 | 0.85 |
| (6,3) | 0.7 | 6 | 0.75 |
| (7,6) | 0.8 | 6 | 0.90 |

The above values are given pre-computed, so that optimized values are chosen for the data transmission path.

As given by MRDPT problem given by Spyros Tragoudas the cost calculations depend on mainly on reliability and poorly on capacity delay and data units.

The MRDPT formula:

$$\alpha(u,v) = -\ln r(uv)^{\left[d(u,v)+\frac{\sigma}{c(u,v)}\right]} \qquad (3.3)$$

**Notations:**

**Table 3.2:** Notations used for the computation of data transmission path

| | |
|---|---|
| G | (V, E); directed acyclic graph or without self-loops |
| P | (V, E, c, d, r, s, t): network |
| V | set of nodes with integer label |
| (u, v) | directed edges |
| r(u, v) | reliability of directed edges |
| d(u, v) | delay of directed edges |
| c(u, v) | capacity of directed edges |
| s | source node |
| t | destination node |
| σ | Units of data to be transmitted through s to t along a path. |
| P(s, t) | path chosen from source to destination node for transmission of data |

As the reliability is the power of capacity and delay, they are multiplicative and thus depended on the reliability. If an application required maximum capacity or minimum delay over reliability, it fails to do so [48].

This MRDPT problem leads us to give multi-constrained data transmission path. But as the network application goes vast and wide the requirement of constrains are become different at every miracle of seconds. So to make it flexible we further step up for tuned multi-constrained data transmission path.

**Proposed formula:**

$$E(u,v) = -\ln r(uv)e^{-\left[d(u,v)+\frac{\sigma}{c(u,v)}\right]} \qquad (3.4)$$

**Further elaboration:**

$$E(u,v) = -\ln r(u,v) + [d(u,v) + \sigma / c(u,v)]$$

$$E(u,v) = K + \alpha L \qquad (3.5)$$

## 3.3 LEMMA FOR QOS

Lemma for the quality of service are explained as below

**3.3.1 Lemma 1:** Reliability of a path is multiplicative of individual edge and is in between of $0 \leq \text{Reliability} \leq 1$ is taken as

$$R(u,v) = \prod_{s}^{t} r(u,v) \qquad (3.6)$$

**3.3.2 Lemma 2:** Capacity of a path is minimum capacity of the edge chosen in the path. It is always taken as minimum for the path.

$$C(u,v) = \min_{s}^{t} c(u,v) \qquad (3.7)$$

**3.3.3 Lemma 3:** Delay of a path chosen for transmission is additive of individual edge taken between the paths from s to t.

$$D(u,v) = \prod_{s}^{t} d(u,v) \qquad (3.8)$$

Let: P is the path chosen from s to t with other vertex consideration P(s, v1, v2... t)

$$C(P) = \min \sum_{i=s}^{t} P_i \qquad (3.9)$$

Numbers of multiple paths are routed and best shortest path has been chosen for data transmission. For the path with data size 5 reliability capacities, delay and new weight are given as below.

**Table 3.3:** Values of the reliability, capacity and delay for the edges of routed path

| Path | Reliability | Capacity | Delay |
|---|---|---|---|
| 0-2 | 0.7 | 0.80 | 7 |
| 2-7 | 0.9 | 1.50 | 7 |
| 7-6 | 0.8 | 0.90 | 6 |
| 6-1 | 0.9 | 0.85 | 5 |
| 1-5 | 0.7 | 0.95 | 4 |

So reliability here is .31, delay is 29 and capacity is .80.

But here not so much focus on delay and capacity so to do this we have introduce a new formula called as MRTDPT.

## 3.4 Problem formulation

Any route chosen can give an optimal path with the feasibility of reliability, capacity and delay. However , if some constraint so aggressive so that no single path or route alone is capable of giving optimal path so to make it optimal tuning factor has been introduced to set up the route optimal with considered factors [49]. For depending on the requirement of application, tuning factor settle down optimally. For delay constraint, if constraint value, say d1, associated with application which is more depending of delay, chosen a route which has delay more than requirement can be controlled by tuning factor. Same tuning is done in case of capacity and reliability [50].

Idea of MRTDPT leads us to take the data transmission depend on all three factors viz. reliability, capacity and delay. The MRTDPT formula uses Dijkstra's shortest path first algorithm to compute the data transmission path from source to destination.

The MRTDPT formula is given as below from equation (3.4) and (3.5) defined the tuning factor with all three constrains where α is the tuning factor. The Tuning factor α helps to take control on three fundamental factors as per requirement of application.

### 3.4.1 Optimum routing

The idea MRTDPT associate with optimum routing due to dependency on factors viz. reliability, capacity and delay. These factors can be tuned with the help of tuning factor. As per the application is so sensitive to delay that tuning factor tuned to give that path which has less delay. Optimum routing gives the best QoS services which are helpful in optimal routing 51.

### 3.4.2 Single Link Weight

Depending on the minimum cost, shortest path is selected using Dijkstra's shortest path first algorithm. Single link weight for routing leads to control optimized traffic form source to destination by considering three major factors for optimization into single link weight of combined effect of all three together [52].

### 3.4.3 Offline and Online Tuning approach

Tuning approach for the optimization of reliability, capacity and delay helps to take care of requirements of the user. Tuning leads to give best QoS services provided by the network. Due to tuning factor real time applications perform well. Tuning is done either by offline or online. Offline tuning is done user by manually as per requirement of user and it takes care of the cost value used for routing by routing protocol. For online tuning, SDN can be used for online tuning approach [53].

## 3.5 Installation and simulation the network simulator (ns-2)

NS2 is the one of the main open source simulator [39]. The main are of using this is in the networking research field. Simulations can be defined as, "estimation of the closeness of the results with the physical scenario with the expected results. The values which lies under validate observations are the final outcome".

### 3.5.1 Overview

NS2 is the open source simulator which is the second version of NS (Network Simulator). Basically NS is based on the REAL network simulator developed at University of California at Berkeley, USA. Now it is a VNIT project supported by DARPA. NS2 is not a fully developed tool for all the research and development. The researcher is responsible for the bugs and error and the validity of their results and share them with the organized community. Various packaged are designed by different organizations or non-profit groups and shared with the world to explore them more. The number of applications is implemented with NS simulator viz. TCP and UDP, Drop tail, FTP and CBR etc.

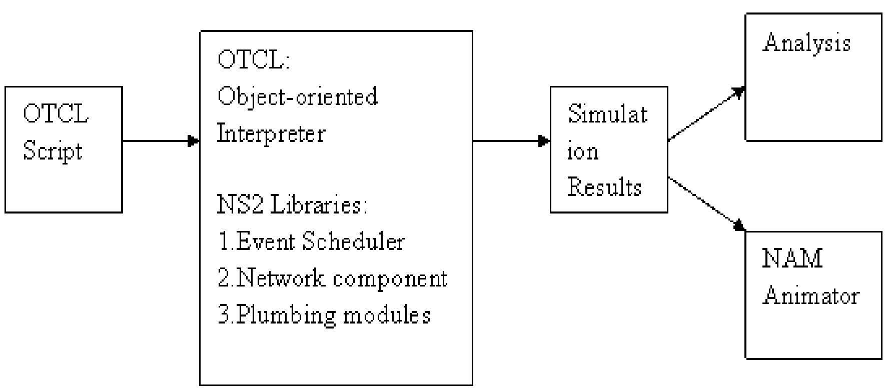



**Figure 3.2:** NS2 tcl script into trace file and NAM file

### 3.5.2 Main features

The main features of the NS2 are that it used two languages C++ and OTcl (Object Oriented extension of Tcl). The main reason to use both these is due to internal characteristics. C++ is efficient in use and faster but not easy to modify and change. The OTcl has the feature which is not deals with C++. OTcl is efficient to simulations scripts. The results taken after simulations of Tcl file are either used for simulation results or either for graphical presentation using Network Animation (NAM).

NS2 is an event driven network simulator, which can be implemented in Linux-based platform. This dissertation will explain how to work with NS2.35 in Ubuntu. To download the NS2 a zipped file is supposed to download from www.isi.edu/nsnam/ns to install it is recommended that don't go with root mode because important file can be affected by this mode [40].

### 3.5.3 Tool Command Language (Tcl)

Tcl is a very powerful and dynamic programming language developed by John Ouster at the University of California, Berkeley. It has very large number of applications like networking, web and desktop applications, testing etc. the main advantage that why we use this language is due to compatibility factor with C++ programming language. The Tcl libraries are directly inter co-related directly into C programs.

### 3.5.4 Network Animation (NAM)

The network animator started in 1990 as a simple tool for animating packet trace data. This is a completely separate program that is distributed with NS simulator to see the flow of packets through the network. NAM generally reads the input file and draws the network graphically.

## 3.6 CONCLUSION

In this chapter the MRDPT problem is discussed and implements in NS-2 simulator. After getting the observation and results the MRDPT problem only plays a significant role in reliability and less concern with the factors capacity and delay. In next chapter we will take proposed cost function and simulate with the help of the NS-2. As a tuning factor gives the best optimized results with all parameters.

# CHAPTER 4

# SIMULATION RESULTS AND DISCUSSION

## 4.1 Results for a given set of values:

For simulations of MRTDPT we have considered traffic size of 5 units and weight factor is based on the value of tuning factor consisting of reliability, capacity and delay. The routing protocol is used Dijkstra's shortest path first algorithm. All these simulation with tuning factor has been done in NS2 simulation tool [12-13].

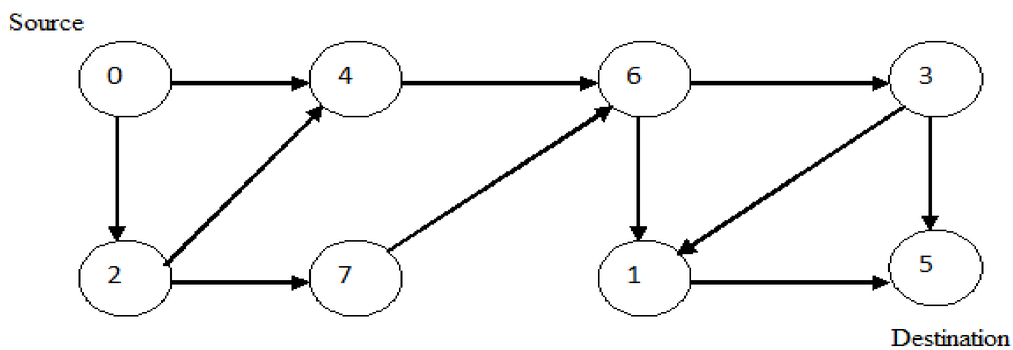



**Figure 4.1:** Topology G (s, t)

As we know that reliability and link delays always changes suddenly at any transmission instant and are not predictable [14]. Our interest is to examine the feasibility of our approximate method of probabilistic modeling of unknown link delay and reliability in wired network. If it achieves well performance for the given values then it will work for well-defined models too.

The performance comparison between two methods MRDPT and MRTDPT is that MRDPT only depends on the reliability and cannot be adjusted as per requirement by application but MRTDPT gives the path selection as per requirement of application.

The network chosen for experiment consist of 7 nodes and 11 edges assigned with different values of reliability, capacity and delay.

**Table 4.1:** Given set of data for topology G (s, t)

| Edge(u,v) | Edge Reliability r(u,v) | Edge Delay d(u,v) | Edge Capacity C(u,v) |
|---|---|---|---|
| (0,2) | 0.7 | 7 | 0.80 |
| (0,4) | 0.7 | 5 | 0.50 |
| (1,5) | 0.7 | 4 | 0.95 |
| (2,4) | 0.8 | 5 | 1.00 |
| (2,7) | 0.9 | 7 | 1.50 |
| (3,1) | 0.8 | 3 | 0.70 |
| (3,5) | 0.7 | 6 | 0.76 |
| (4,6) | 0.6 | 6 | 0.80 |
| (6,1) | 0.9 | 5 | 0.85 |
| (6,3) | 0.7 | 6 | 0.75 |
| (7,6) | 0.8 | 6 | 0.90 |

The table for the equation 1 has been given below so that routing is done from source to destination. Numbers of different paths are there available there from source node to destination. Let P(s, t) donate the path set of P possible path from a source node s to t. Each path is associated with delay d(u, v), reliability r(u, v) and capacity c(u, v). As the combined values of these three factors gives optimal path for routing with the tuning factor alpha (α) with the different values of alpha are: 1, ½. 1/3, ¼, 1/10, 1/20, 1/30, 1/40 and 1/100.

.

**Table 4.2:** Cost calculated depending on the tuning factor with given set of values

| α =1 | α =1/2 | α =1/3 | α =1/4 | α =1/10 | α =1/20 | α =1/30 | α =1/40 | α =1/100 |
|---|---|---|---|---|---|---|---|---|
| 13.60 | 6.97 | 4.76 | 3.66 | 1.67 | 1.01 | 0.79 | 0.68 | 0.48 |
| 15.35 | 7.85 | 5.35 | 4.10 | 1.85 | 1.1 | 0.85 | 0.72 | 0.5 |
| 9.61 | 4.98 | 3.43 | 2.66 | 1.27 | 0.81 | 0.65 | 0.58 | 0.44 |
| 10.22 | 5.22 | 3.55 | 2.72 | 1.22 | 0.72 | 0.55 | 0.47 | 0.32 |
| 1.43 | 5.26 | 3.54 | 2.68 | 1.13 | 0.61 | 0.44 | 0.35 | 0.20 |
| 10.36 | 5.29 | 3.60 | 2.75 | 1.23 | 0.72 | 0.55 | 0.47 | 0.32 |
| 12.92 | 6.63 | 4.54 | 3.49 | 1.6 | 0.27 | 0.76 | 0.66 | 0.47 |
| 12.76 | 6.63 | 4.59 | 3.57 | 1.73 | 1.12 | 0.91 | 0.82 | 0.64 |
| 10.98 | 5.54 | 3.92 | 2.82 | 1.18 | 0.64 | 0.46 | 0.37 | 0.21 |
| 13.01 | 6.68 | 4.5 | 3.5 | 1.61 | 0.98 | 0.77 | 0.66 | 0.48 |
| 11.77 | 5.99 | 4.07 | 3.10 | 1.37 | 0.79 | 0.60 | 0.50 | 0.34 |
| Path chosen for different values of alpha | | | | | | | | |
| 0-4-6-1-5 | 0-4-6-1-5 | 0-4-6-1-5 | 0-4-6-1-5 | 0-4-6-1-5 | 0-4-6-1-5 | 0-4-6-1-5 | 0-2-7-6-1-5 | 0-2-7-6-1-5 |

As the different values of cost is given with respect to different values of alpha α so that routes are chosen as per requirement of application.

For the path chosen for routing 0-4-6-1-5 has reliability value 0.26, capacity value .50 and delay value 20 sec.

**Table 4.3:** Values of the reliability, capacity and delay for the edges of routed path 0-4-6-1-5

| Path | Reliability | Capacity | Delay |
|---|---|---|---|
| 0-2 | 0.7 | 0.80 | 7 |
| 2-7 | 0.9 | 1.50 | 7 |
| 7-6 | 0.8 | 0.90 | 6 |
| 6-1 | 0.9 | 0.85 | 5 |
| 1-5 | 0.7 | 0.95 | 4 |

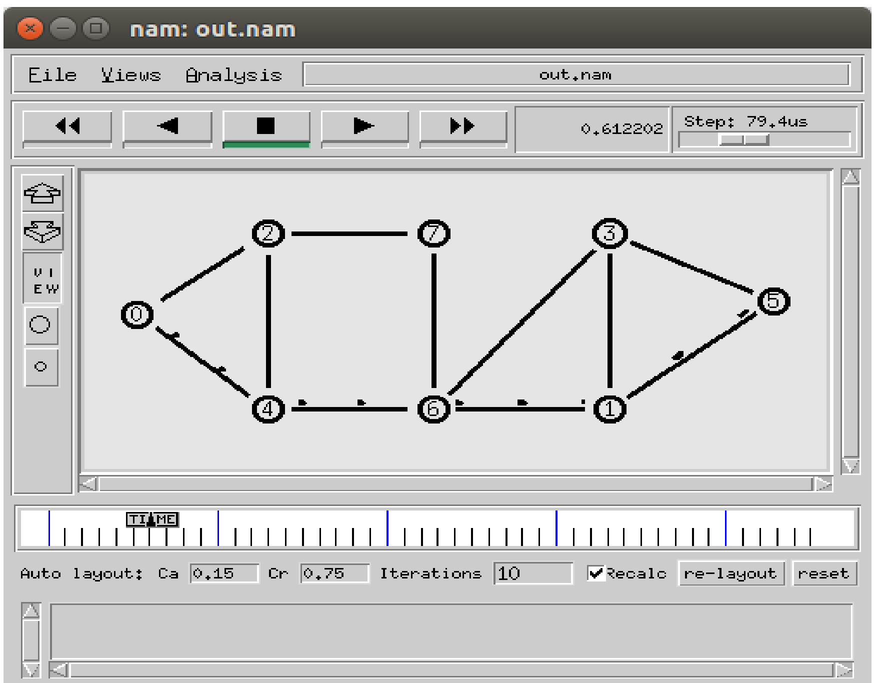



**Figure 4.2:** Path chosen for routing 0-4-6-1-5 with tuning factor of 1/3

The value of alpha has been controlled the different values of reliability, capacity and delay are given. With variation of value of alpha path from source node to destination node also changes to give the optimized route with respect to reliability, delay and capacity.

And for the path chosen for routing 0-2-7-6-1-5 has reliability value 0.31, capacity value .80 and delay value 29 sec.

**Table 4.4:** Values of the reliability, capacity and delay for the edges of routed path 0-2-7-6-1-5

| Path | Reliability | Capacity | Delay |
|---|---|---|---|
| 0-2 | 0.7 | 0.80 | 7 |
| 2-7 | 0.9 | 1.50 | 7 |
| 7-6 | 0.8 | 0.90 | 6 |
| 6-1 | 0.9 | 0.85 | 5 |
| 1-5 | 0.7 | 0.95 | 4 |

Graphical output from the NAM is taken below to gives the shortest path with defined st of values of reliability, capacity and delay.

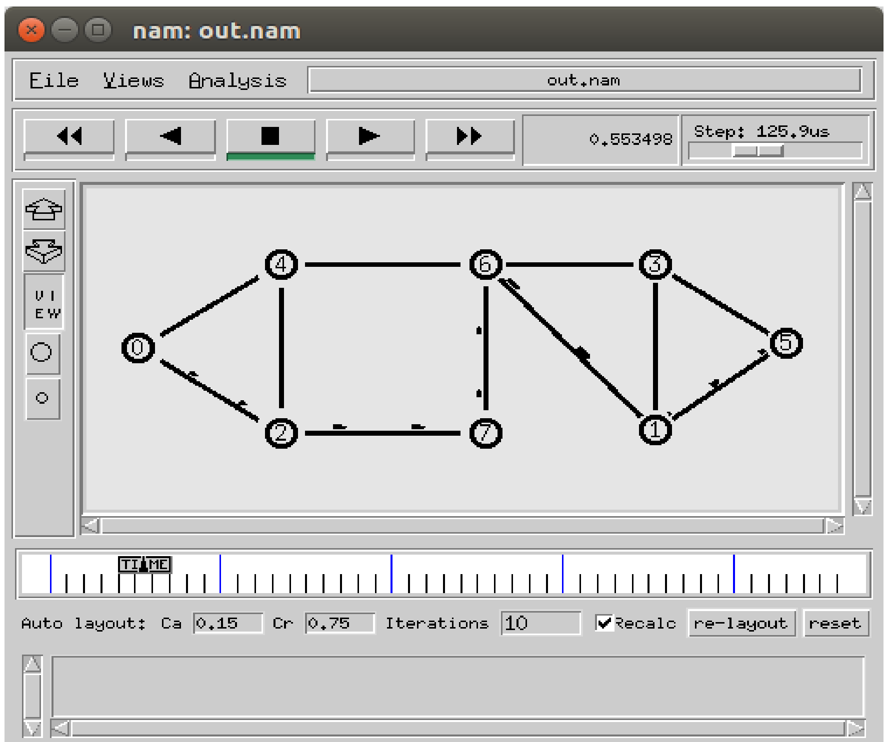



**Figure 4.3:** Path chosen for routing 0-2-7-6-1-5 with tuning factor of 1/100

So as per requirement of application, user can set value of tuning factor. Such as for real time application user can set alpha value such that delay is minimum, so that user can select path 0-4-6-1-5 and for traffic network where user needs of high bandwidth, can use path 0-2-7-6-1-5 to get high capacity and reliability.

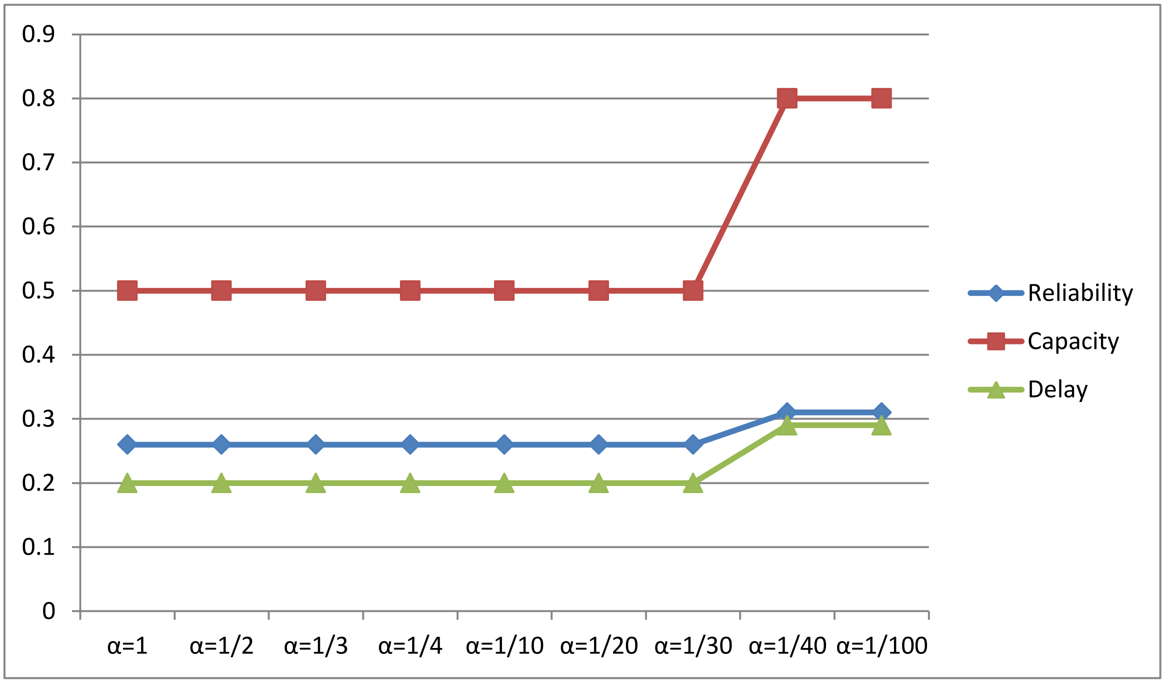



**Figure 4.4:** Figure of values of constraints for different values of alpha

## 4.2 Results for the random set of values:

As it is known that the network environment is random in nature so the experiment results must be validated by operating network in random environment. The change in environment is random sometime it is shiny, rainy and dry so due to changes in environment the certain factor got changed. The random values for all links are taken with respect to capacity, delay and reliability (See APENDIX B). The random values generated for simulation are derived using java language on Netbeen tool. Other platforms can be used for the generation of random values viz. Matlab, c and C++. The listed table is given below with the random values for different values of tuning factor alpha.

The topology chooses the best path with tuning factor and user can use that tuning factor according to the required applications. To finding the best path with tuning factor is purely based on hit and trial method and this hypothesis may be sometime become NP-Complete problem for the big topology. The main aim of the tuning factor is just do the trade-off with the constraints and give the best optimized path. It may not necessary always that path will be optimized with all constraints; sometimes it may be worst with respect to other parameter and best with required parameter.

Table generated with random values to the links is given below to analysis the best path with application.

**Table 4.5:** Random set of data for topology G (s, t)

| Edge(u,v) | Edge Reliability r(u,v) | Edge Delay d(u,v) | Edge Capacity C(u,v) |
|---|---|---|---|
| (0,2) | .89 | 1 | 7.9 |
| (0,4) | .43 | 0 | 2.5 |
| (1,5) | .56 | 4 | 4.52 |
| (2,4) | .7 | 6 | 4.38 |
| (2,7) | .75 | 4 | 4.09 |
| (3,1) | .98 | 6 | 4.68 |
| (3,5) | .24 | 5 | 7.54 |
| (4,6) | .01 | 2 | 8.52 |
| (6,1) | .13 | 7 | 2.98 |
| (6,3) | .98 | 5 | 1.9 |
| (7,6) | .94 | 2 | 8.06 |

Tuning factor used to get trade-off between the constraints and different values chosen for the tuning factor are 5, 2, 1, 0.1, 0.5, 0.25, 0.02, 0.0125, 0.005 and 0.002.

Detailed tables for the different path chosen by the traffic with the different values of tuning factor are given below.

**Table 4.6:** Cost calculated depending on the tuning factor with random values generated

| α =5 | α =2 | α =1 | α =.1 | α =0.5 | α =0.25 | α =.02 | α =.0125 | α =.005 | α =0.002 |
|---|---|---|---|---|---|---|---|---|---|
| 8.28 | 3.38 | 1.79 | .27 | .93 | 0.52 | 0.14 | 0.13 | 0.12 | 0.11 |
| 10.84 | 4.84 | 2.84 | 1.04 | 1.84 | 1.34 | .88 | 0.86 | 0.85 | 0.84 |
| 26.11 | 10.79 | 5.68 | 1.09 | 3.13 | 1.85 | 0.68 | 0.64 | 0.60 | 0.59 |
| 36.06 | 14.63 | 7.49 | 1.07 | 3.92 | 2.14 | 0.49 | 0.44 | 0.39 | 0.37 |
| 26.40 | 10.73 | 5.51 | .80 | 2.89 | 1.59 | 0.39 | 0.35 | 0.31 | 0.29 |
| 35.36 | 14.15 | 7.08 | .72 | 3.55 | 1.78 | 0.16 | 0.10 | 0.05 | 0.03 |
| 29.74 | 12.75 | 7.09 | 1.99 | 4.25 | 2.84 | 1.54 | 1.49 | 1.45 | 1.43 |
| 17.53 | 9.77 | 7.19 | 4.86 | 5.89 | 5.25 | 4.65 | 4.63 | 4.61 | 4.61 |
| 45.42 | 19.39 | 10.71 | 2.90 | 6.37 | 4.20 | 2.21 | 2.14 | 2.08 | 2.05 |
| 38.17 | 15.28 | 7.65 | 0.78 | 3.83 | 1.92 | 0.17 | 0.11 | 0.05 | 0.03 |
| 13.16 | 5.30 | 2.68 | 0.32 | 1.37 | 0.71 | 0.11 | 0.09 | 0.07 | 0.06 |
| Path chosen for different values of alpha | | | | | | | | | |
| 0-4-6-3-5 | 0-4-6-3-5 | 0-2-7-6-3-5 | 0-2-7-6-3-1-5 | 0-2-7-6-3-5 | 0-2-7-6-3-5 | 0-2-7-6-3-1-5 | 0-2-7-6-3-1-5 | 0-2-7-6-3-1-5 | 0-2-7-6-3-1-5 |

Using different random values for the given topology with different values of tuning factor, there are three different paths have been chosen for the data transmission.

For the path chosen for routing 0-4-6-3-5 has reliability value 0.001, capacity value 1.9 and delay value 12 sec. Here the path chosen has very less reliability but as we see with respect to delay constraint it is best.

**Table 4.7:** Values of the reliability, capacity and delay for the edges of routed path 0-4-6-3-5

| Path | Reliability | Capacity | Delay |
|---|---|---|---|
| 0-4 | 0.43 | 2.5 | 0 |
| 4-6 | 0.01 | 8.52 | 2 |
| 6-3 | 0.98 | 1.9 | 5 |
| 3-5 | 0.24 | 7.54 | 5 |

The graphical view in the NAM is shown as in figure with the path 0-4-6-3-5.

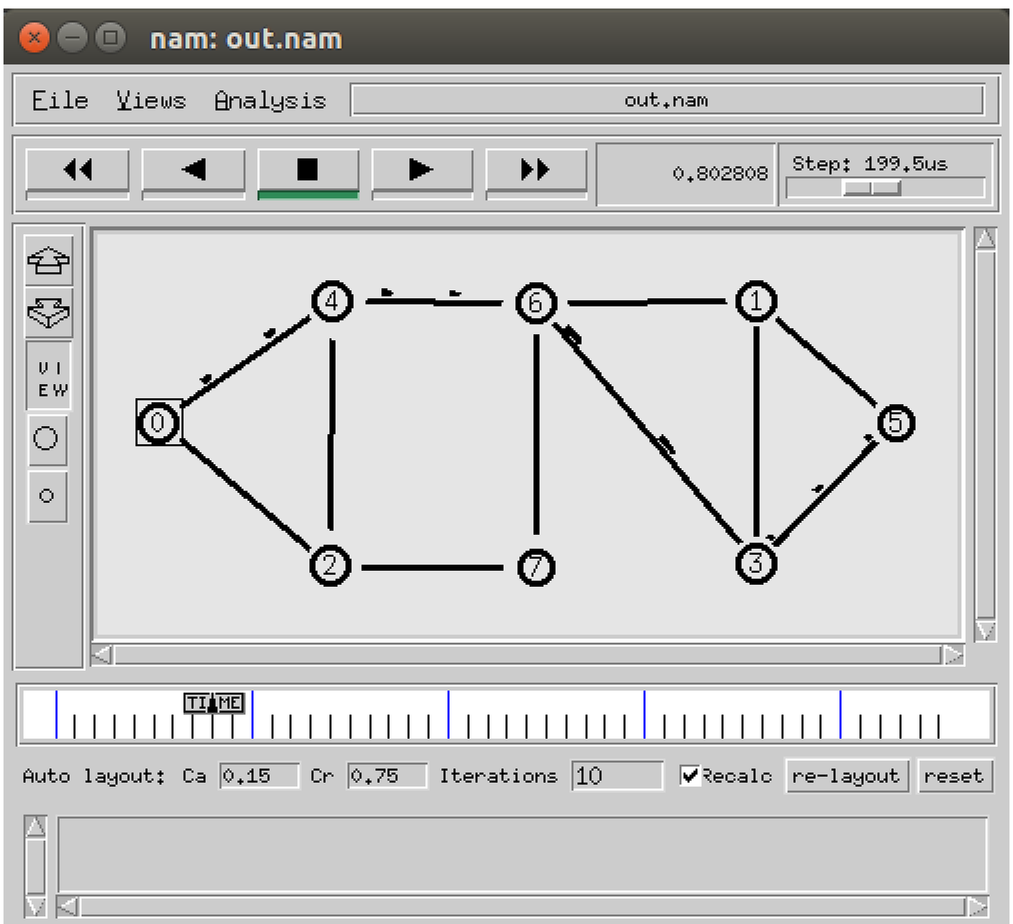



**Figure 4.5:** Path chosen for routing 0-4-6-3-5 with tuning factor of 2

And for the path chosen for routing 0-2-7-6-3-1-5 has reliability value 0.32, capacity value .1.9 and delay value 22 sec.

**Table 4.8:** Values of the reliability, capacity and delay for the edges of routed path 0-2-7-6-3-1-5 with given **values**

| Path | Reliability | Capacity | Delay |
|---|---|---|---|
| 0-2 | 0.89 | 7.9 | 1 |
| 2-7 | 0.75 | 4.09 | 4 |
| 7-6 | 0.94 | 8.06 | 2 |
| 6-3 | 0.93 | 1.9 | 5 |
| 3-1 | 0.98 | 4.68 | 6 |
| 1-5 | .56 | 4.52 | 4 |

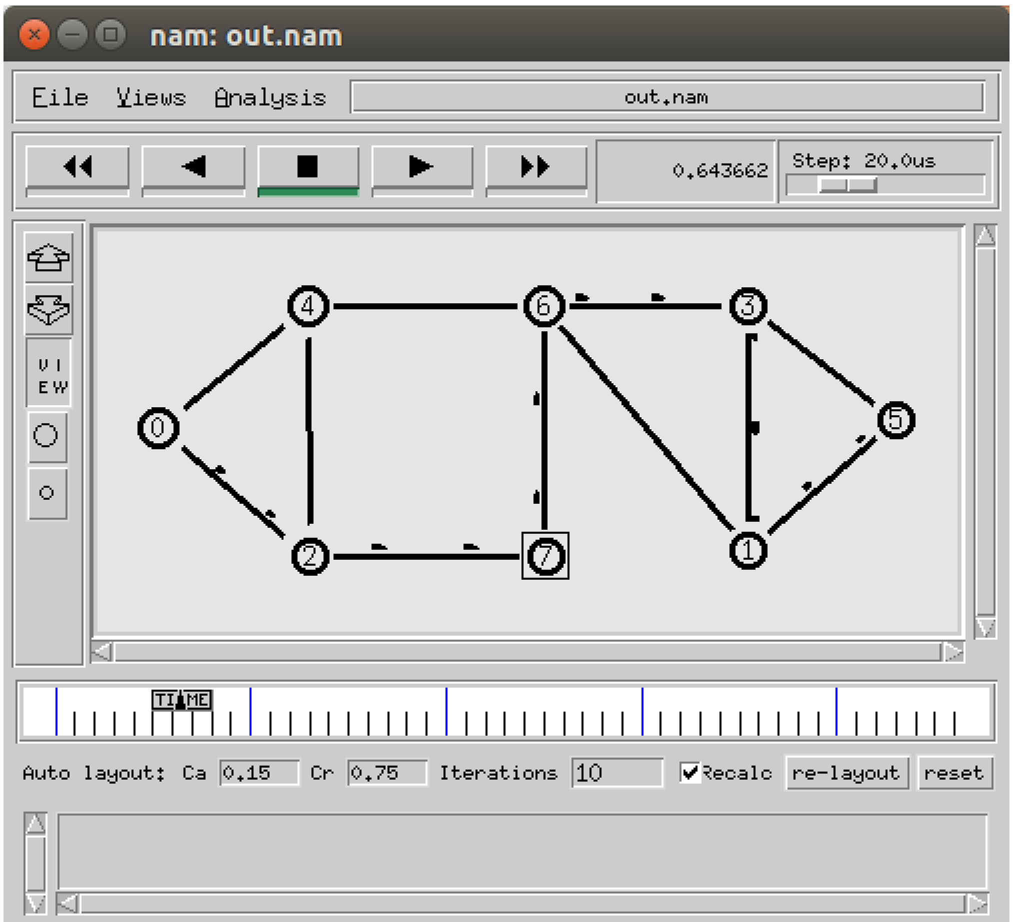



**Figure 4.6:** Path chosen for routing 0-2-7-6-3-1-5 with tuning factor of 0.1

And for the path chosen for routing 0-2-7-6-3-5 has reliability value 0.14, capacity value 1.9 and delay value 17 sec.

**Table 4.9:** Values of the reliability, capacity and delay for the edges of routed path 0-2-7-6-3-5

| Path | Reliability | Capacity | Delay |
|---|---|---|---|
| 0-2 | 0.89 | 7.9 | 1 |
| 2-7 | 0.75 | 4.09 | 4 |
| 7-6 | 0.94 | 8.06 | 2 |
| 6-3 | 0.93 | 1.9 | 5 |
| 3-5 | 0.24 | 7.54 | 5 |

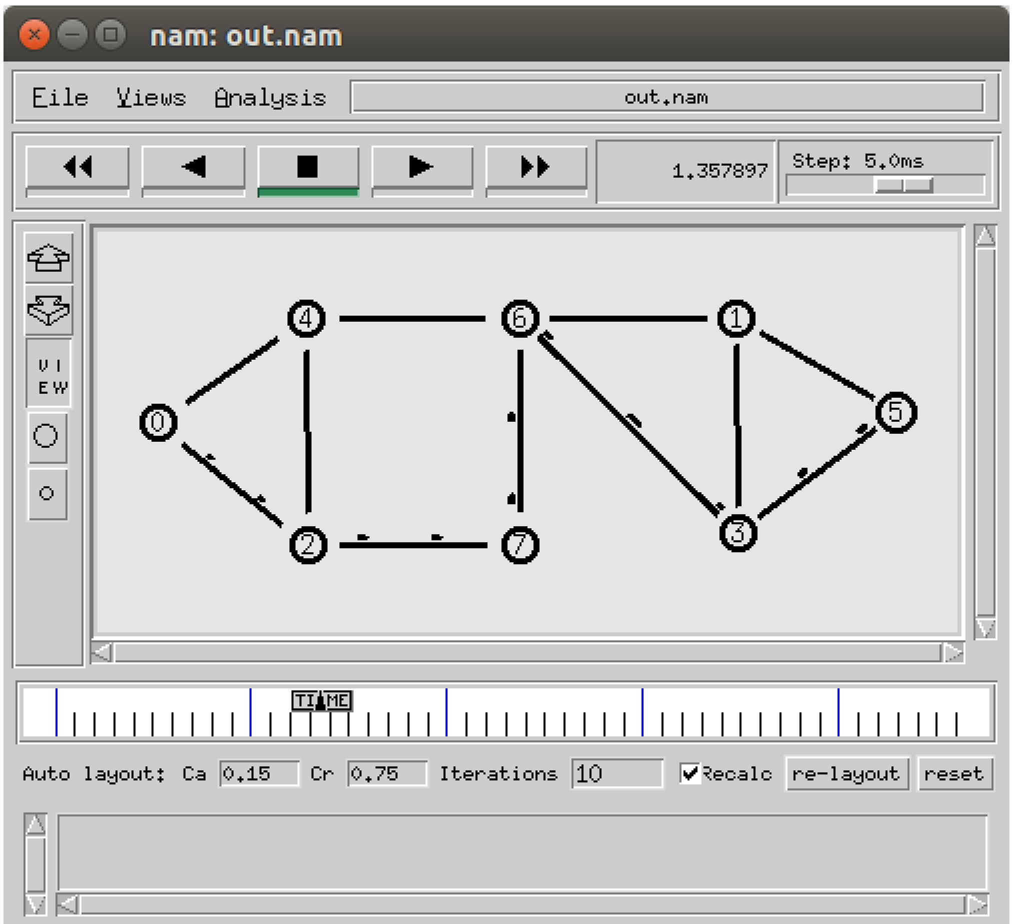



**Figure 4.7:** Path chosen for routing 0-2-7-6-3-1-5 with tuning factor of 0.5

The graph for all values that we got from different path chosen with the different values of alpha has been shown below.

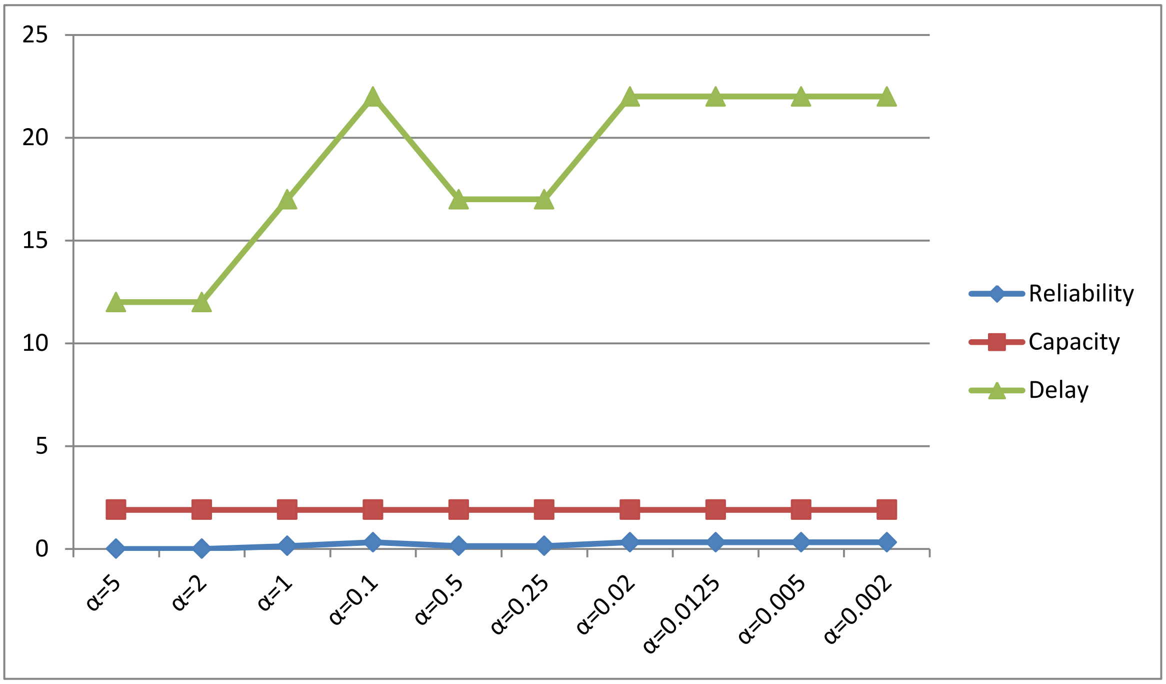



**Figure 4.8:** Figure of values of constraints for different values of alpha

## 4.3 CONCLUSION

This chapter concludes that the process of finding optimized path is a NP hard problem but still one can find optimized results. As it is known that network environment is random in nature so set of random values also associated with topology to check the compatibility of results.

Using random values anyone can go to one of the amongest three paths which are optimized with the concern factor.

# CHAPTER 5

# CONCLUSION AND FUTURE DIRECTION

In this dissertation, we proposed a formula MRTDPT which computes the link routes on three main constraints viz. reliability, capacity and delay. The existing formula MRDPT, just consider one constraint namely reliability and not holds on capacity and delay as MRTDPT does flexible approach to reliability, capacity and delay. A simulation result carried here are depend on the given set of values of reliability, capacity and delay and shows its feasibility towards the optimality of routes give the optimal results with the application requirements. Further also tuning done here is offline but one can do the tuning online using Software Defined Network.

# REFERENCE LIST

[1] D. E. Comer, Internetworking with TCP/IP, 3rd ed: Prentice-Hall, 1995, vol. I

[2] Q. Ma and P. Steenkiste, “Quality of service routing for traffic with performance guarantees,” in Proc. 4th Int’l. IFIP Workshop on Quality of Service, 1997.

[3] T. Korkmaz and M. Krunz, "Multi-Constrained Optimal Path Selection", lEE INFOCOM 2001.

[4] P. Stavroulakis, Reliability, Survivability and Quality of Large Scale Telecommunication System, John Wiley and Sons, Ltd., U.K., 2003.

[5] R. K. Ahuja, J. L. Batra, and S. K. Gupta, “Combinatorial optimization with rational objective functions: A communication,” Mathematics Operations Res., vol. 8, p. 314, 1983.

[6] F.Kuiper, T.Korkmaz and M. Krunz, " An Overview of Constraint-Based Path Selection Algorithms for QoS Routing", IEEE Communication Magazine, Dec. 2002, pp 5055.

[7] R. Kumar, K. Gopal, "An Algorithm for Computing the Best-Performing Path in a Computer Network, " International Journal of Performability Engineering, Volume 3, Number 2, pp.203-212, 2007.

[8] http://www.l-com.com/content/Article.aspx?Type=N&ID=10431

[9] M. Pioro and D. Medhi, Routing, Flow, and Capacity Design in Communication and Computer Network, Morgan Kaufinann Publishers, N. Delhi, 2005.

[10] S. Savage, A. Collins and E. Hoffman, "The End-to-End Effects of Internet Path Selection", Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication, pp289-299, 1999.

[11] A. Chakrabarti and G. Manimaran, "Reliability Constrained Routing in QoS Networks", IEEE/ACM Trans. Networking, vol. 14, no. 3, June 2005, pp 662-675.

[12] Z.Wang and J. Crowcroft, “Bandwidth-delay based routing algorithms,” inProc. GLOBECOM’1995, vol. 3, 1995, pp. 21–2133.

[13] F.-S. P. Tsen, T.-Y. Sung, and M.-Y. Lin et al., “Finding the most vital edges with respect to the number of spanning trees,” IEEE Trans. Reliability, vol. 43, no. 4, pp. 600–602, 1994.

[14] Banerjee, G. and Sidhu, D. (2002). Comparative analysis of path computation techniques for MPLS traffic engineering. Computer Netw~orks4, 0: 149-165.

[15] S. Avallone, F.A. Kuipers, G. Ventre, and P. Van Mieghem. Dynamic routing in QoS Traffic Engineered networks. In EUNICE IFIP WG 6.6, WG 6.4 and WG 6.9 Workshop, pages 222–228, Colmenarejo, Spain, Jul 2005.

[16] Kuipers, F., Van Mieghem, P., Korkniaz, T., and Krunz, M. (2002). An overview of constraintbased path selection algorithms for QoS routing. IEEE Commzrnications Magazine, 40(12):50-55.

[17] S. Tragoudas, "The most Reliable Data-Path Transmission", IEEE Trans. Reliability, Vol. 50, No.3, Sept. 2001.

[18] Cormen, T. H., C. E. Leiserson, R. L. Rivest, and C. Stien, Introduction to Algorithms, Prantice-Hall of India Pvt. Ltd., N. Delhi, 2002.

[19] M. Pioro and D. Medhi, Routing, Flow, and Capacity Design in Communication and Computer Network, Morgan Kaufinann Publishers, N. Delhi, 2005.

[20] Andrew S. Tanenbaum, "Computer Networks 4Th Ed.", Prentice Hall, 2003 .

[21] Hayes, Graves, Mullett, Chipps, Wired, Optical and Wireless communication paperback – 2008.

[22] http://www.codeproject.com/Articles/999691/RESTful-Day-sharp-Custom-URL-Re-Writing-Routing-us

[23] Jeff Doyle, Jennifer DeHaven Carroll Routing TCP/IP, Volume II (CCIE Professional Development) Published Apr 11, 2001 by Cisco Press.

[24] Clausen, T., Jacquet, P., Laouiti, A., Muhlethaler, P., Qayyum, A., Viennot, L.: Optimized Link State Routing Protocol. Experimental RFC3626, version 11, IETF, October (2003).

[25] P. Kuppusamy, D. Thirunavukkarasu dan D. Kalaavathi, "A Study and Comparison of OLSR, AODV and TORA Routing Protocols in Ad Hoc Networks," IEEE International Symposium on Public, 2011.

[26] Asma Tuteja ,Rajneesh Gujral Mullana, Sunil Thalia "Comparative Performance Analysis of DSDV, AODV and DSR Routing Protocols in MANET using NS2", International Conference on Advances in Computer Engineering, 2010IEEE.

[27] Y. Zheng, J Tian, Z. Liu and W. Dou, "ALimitedPath Unicast QoS Routing Algorithm", ISCC 04: Ninth IEEE Symposium on Computers and Communications.

[28]http://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus3000/sw/unicast/503_u1_2/nexus3000_unicast_config_gd_503_u1_2/l3_overview.html

[29] S. Bae , S. Lee and M. Gerla "Unicast performance analysis of extended ODMRP in a wired-to-wireless hybrid ad-hoc network", Proc. Mil. Commun. Conf., pp.1228 -1232 2002.

[30] E. Dijkstra "A note on two problems in connexion with graphs", Numerische mathematik, vol. 1, no. 1, pp.269 -271 1959.

[31] Li Layuan, Li Chunlin, Yaun Peiyan Performance evaluation and simulations of routing protocols in ad hoc networks.

[32] M. I. Henig, "The shortest path problem with two objective functions", Eur. J. Oper. Res, vol 25.

[33] Cavendish, Dirceu, and Mario Gerla. "Internet QoS routing using the Bellman-Ford algorithm." High Performance Networking. Springer US, 1998.

[34] M. Portmann and A. A. Pirzada "Wireless mesh networks for public safety and crisis management applications", Internet Computing, IEEE, vol. 12, no. 1, pp.18 -25 2008.

[35] J. McCabe, Network analysis, architecture, and design. Morgan Kaufmann, 2007.

[36] Chowdhury, M.U.; Perera, D.; Pham, T.; , "A performance comparison of three wireless multi hop ad-hoc network routing protocols when streaming MPEG4 traffic," Multitopic Conference, 2004. Proceedings of INMIC 2004. 8th International , vol., no., pp. 516- 521, 24-26 Dec. 2004. doi: 10.1109/INMIC.2004.1492933

[37] H. Ehsan and Z. A. Uzmi (2004), "Performance Comparison of Ad HocWireless Network Routing Protocols", IEEE 8th International Multitopic Conference, Proceedingsof INMIC, pp.457 – 465, December 2004.

[38] C. Perkins, E. B. Royer, S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing - Internet Draft", RFC 3561, IETF Network Working Group, July 2003.

[39] ]Issariyakul, T., & Hossain, E. (2012). An introduction to network simulator NS2. New York: Springer.

[40] The Network Simulator vers.2.26 (NS2) Home Page www.isi.edu/nsnam/ns

[41] M. Liotine, Mission-Crritical Network Planning, Artech House, Inc., London, 2003.

[42] Soh S., Rai S., "An Efficient Approach for Evaluating Communication-Network Reliability with Heterogeneous Link-Capacities," IEEE Trans. on Rel., vol. 54, no.1, pp. 133-142, March 2005.

[43] Jeff Doyle, Jennifer DeHaven Carroll Routing TCP/IP, Volume II (CCIE Professional Development) Published Apr 11, 2001 by Cisco Press.

[44] Ashutosh Sharma, Rajiv Kumar, “Realistic Comparison of Performance Parameters of Static and Dynamic Unicast Routing over Mesh Topology,” International Journal of Scientific & Engineering Research (IJSER), Volume 6, Issue 12,Dec 2015.

[45] Cormen, T. H., C. E. Leiserson, R. L. Rivest, and C. Stien, Introduction to Algorithms, Prantice-Hall of India Pvt. Ltd., N. Delhi, 2002.

[46] JD. McCABE, Network analysis, architecture and design, 2ndedition, Morgan Kaufmann Publication, 2003.

[47] G. Xue, "End-to-End Data Paths: Quickest or Most Reliable?", IEEE Communications Letters, Vol. 2, No.6, pp. 156-158, 1998.

[48] P. Stavroulakis, Reliability, Survivability and Quality of Large Scale Telecommunication System, John Wiley and Sons, Ltd., U.K., 2003.

[49] S. Chen and K. Nahrstedt, “On finding multi-constrained paths,” in Proc. ICC’1998, pp. 874–879.

[50] R. Luus, “Application of iterative dynamic programming to optimal control ofnonseparable problems,” Hung. J. Industrial Chemistry, vol. 25, pp. 293–297, 1997.

[51] Huang, Xiaoxia, and Yuguang Fang. "Multiconstrained QoS multipath routing in wireless sensor networks." Wireless Networks 14.4 (2008): 465-478.

[52] M. Pioro and D. Medhi, Routing, Flow, and Capacity Design in Communication and Computer Network, Morgan Kaufinann Publishers, N. Delhi, 2005.

[53] Mendiola, Alaitz, et al., "DynPaC: a path computation framework for SDN", Software Defined Networks (EWSDN), 2015 Fourth European Workshop on. IEEE, 2015.

# APENDIX

**A.1** CODE FOR GENRATION OF RANDOM NUMBER OF RELIABILITY, CAPACITY AND DELAY FOR THE LINKS

```
package random;
import java.util.Scanner;
/**
 *
 * @author ashutosh_java
 */
public class random {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        float r[];
        r =new float[57];

        int d[];
        d =new int[57];

        float c[];
        c =new float[57];

        float a[];
        a =new float[10];
        float e[];
        e =new float[57];

        Scanner cs =new Scanner(System.in);
        System.out.println("enter value");
        for(int i=0;i<10;i++)
```

```
        {
           a[i]=cs.nextFloat();
        }

        for(int j=0;j<57;j++)
        {
           r[j]=((int)(Math.random()*100));
           r[j]=r[j]/100;

           d[j]=((int)(Math.random()*100));
           d[j]=d[j]/10;

           c[j]=((int)(Math.random()*1000));
           c[j]=c[j]/100;
        }
        System.out.println("reliabilty    delay      capcity        log            alfa            cost");
        for(int i=0;i<10;i++)
        {
           for(int j=0;j<57;j++)
           {
              float l;
              l=(float) -Math.log(r[j]);
              e[j]=(float) l+(a[i]*(d[j]+57/c[j]));
              System.out.println(r[j]+"              "+d[j]+"            "+c[j]+"              "+l+"            "+a[i]+"
"+e[j]);
           }
           System.out.println("\n\n");
        }
    }
}
```

**B.1** CODE FOR NETWORK TOPOLOGY WITH COST CORSPONDING TO ALPHA 5

```
set ns [new Simulator]
#To create the trace files we write
set tracefile1 [open out.tr w]
$ns trace-all $tracefile1

#To create the nam files we write
set namfile1 [open out.nam w]
$ns namtrace-all $namfile1

# Defining the 'finish' procedure'
proc finish {} {
        global ns namfile1
        global ns tracefile1
        $ns flush-trace
        close $tracefile1
        close $namfile1
        exec nam out.nam &
        exit 0
}
#create nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
set n6 [$ns node]
set n7 [$ns node]

#create links between the nodes
$ns duplex-link $n0 $n2 10Mb 10ms DropTail
$ns duplex-link $n0 $n4 10Mb 10ms DropTail
$ns duplex-link $n1 $n5 10Mb 10ms DropTail
```

```
$ns duplex-link $n2 $n4 10Mb 10ms DropTail
$ns duplex-link $n2 $n7 10Mb 10ms DropTail
$ns duplex-link $n3 $n1 10Mb 10ms DropTail
$ns duplex-link $n3 $n5 10Mb 10ms DropTail
$ns duplex-link $n4 $n6 10Mb 10ms DropTail
$ns duplex-link $n6 $n1 10Mb 10ms DropTail
$ns duplex-link $n6 $n3 10Mb 10ms DropTail
$ns duplex-link $n7 $n6 10Mb 10ms DropTail
#set queue-size of the link (n2-n3) to 20
#$ns queue-limit $n2 $n3 20

#Create a UDP agent and attach it to node n0
set udp0 [new Agent/UDP]
$udp0 set class_ 1
$ns attach-agent $n0 $udp0

#Initiating FTP over TCP
#set ftp [new Application/FTP]
#$ftp attach-agent $tcp

# Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 1024
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

#Create a Null agent (a traffic sink) and attach it to node n6
set null0 [new Agent/Null]
$ns attach-agent $n5 $null0

#Connect the traffic source with the traffic sink
$ns connect $udp0 $null0

#Schedule events for the CBR agents
```

```
$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"

#Give position to the nodes in NAM
#$ns duplex-link-op $n0 $n1 orient-left
#$ns duplex-link-op $n0 $n2 orient-left
#$ns duplex-link-op $n2 $n5 orient-down
#$ns duplex-link-op $n1 $n3 orient-up
#$ns duplex-link-op $n5 $n6 orient-right
#$ns duplex-link-op $n3 $n6 orient-right

$ns cost $n0 $n2 8.2
$ns cost $n0 $n4 10.84
$ns cost $n1 $n5 26.11
$ns cost $n2 $n4 36.06
$ns cost $n2 $n7 26.40
$ns cost $n3 $n1 35.36
$ns cost $n3 $n5 29.74
$ns cost $n4 $n6 17.53
$ns cost $n6 $n1 45.42
$ns cost $n6 $n3 38.17
$ns cost $n7 $n6 13.16
#Call the finish procedure after 5 seconds of simulation time
$ns at 5.0 "finish"
#Run the simulation
$ns run
```

# LIST OF PUBLICATIONS

1. **Ashutosh Sharma**, Rajiv Kumar, **"Realistic Comparison of Performance Parameters of Static and Dynamic Unicast Routing over Mesh Topology"** in *International Journal of Scientific & Engineering Research (IJSER)*, Volume 6, Issue 12, Dec 2015. doi:10.14299/ijser.2015.12.003
2. **Ashutosh Sharma**, Rajiv Kumar, Poonam Koundal **"A Tuning-based Approach for the Multi-Constrained Data-Path Transmission"** in *International Journal of Control Theory and Applications* (**Accepted (Scopus Indexed)**)
3. **Ashutosh Sharma**, Rajiv Kumar, **"Performance Comparison and Detailed Study of AODV, DSDV, DSR, TORA and OLSR Routing Protocols in Ad Hoc Networks"** in *5th Edition of International Conference on Wireless Networks and Embedded Systems.* **(Communicated)**
4. **Ashutosh Sharma,** Rajiv Kumar **"A Tunned Multiconstrained Link Weight Assignment Algorithm for Optimized Data Transmission"** *IJES, Inderscience*. **(Communicated)**