**Shortest path routing**

A Project Report

Submitted in the partial fulfillment of the requirements for the award of the degree of

# Bachelor of Technology in

Department of Computer Science and Engineering

By

L. Vamsi krishna (2010030410)

V. Abhiram (2010030383)

T. Sainath (2010030174)

under the supervision of

# **Dr. P Lalitha Surya Kumari**

# Professor



Department of Computer Science and Engineering

K L University Hyderabad,

Aziz Nagar, Moinabad Road, Hyderabad – 500 075, Telangana, India.

March, 2022

**Declaration**

The Project Report entitled “Shortest Path Routing” is a record of bonafide work of (L. Vamsi Krishna) (V. Abhiram) (T. Sainath ) submitted in partial fulfillment for the award of B.Tech in the Department of Computer Science and Engineering to the K L University, Hyderabad. The results embodied in this report have not been copied from any other Departments/University/Institute.

L.Vamsi Krishna

T. Sainath

V. Abhiram

**Certificate**

This is to certify that the Project Report entitled “SHORTEST PATH BETWEEN NODES” is being submitted by L. Vamsi Krishna (2010030410), T. Sainath (2010030174), V. Abhiram (2010030383) submitted in partial fulfillment for the award of B.Tech in CSE to the K L University, Hyderabad is a record of bonafide work carried out under our guidance and supervision.

The results embodied in this report have not been copied from any other departments/ University/Institute.

## Signature of the Supervisor

# 

## Signature of the HOD Signature of the External Examine

**ACKNOWLEDGEMENT**

First and foremost, we thank the lord almighty for all his grace & mercy showered upon us, for completing this project successfully.

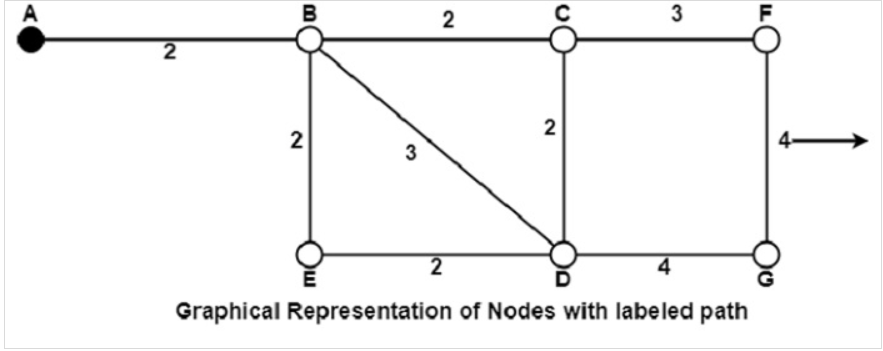
We take grateful opportunity to thank our beloved Founder and Chairman who has given constant encouragement during our course and motivated us to do this project. We are grateful to our Principal **Dr. L. Koteswar Rao** who has been constantly bearing the torch for all the curricular activities undertaken by us.

We pay our grateful acknowledgement & sincere thanks to our Head of the Department **Dr. Chiranjeevi Manike** for his exemplary guidance, monitoring and constant encouragement throughout the course of the project. We thank **Dr. Lalitha Surya Kumari** of our department who has supported throughout this project holding a position of supervisor.

We whole heartedly thank all the teaching and non-teaching staff of our department without whom we won’t have made this project a reality. We would like to extend our sincere thanks especially to our parent, our family members and friends who have supported us to make this project a grand success.

**ABSTRACT**

Routing is the act of moving information across a network from a source to a destination. Along the way, at least one intermediate node typically is encountered. Routing is often contrasted with bridging, which might seem to accomplish precisely the same thing to the casual observer. Routing involves two basic activities: determining optimal routing paths and transporting information groups through a network. Routing also refers to path finding between source and destination. This literature review investigates some of the gateways to path finding in different networks that are listed in present research literature. A selected set of different approaches are highlighted and set in a broader context, illustrating the various aspects of path finding in different networks. Because path finding is applicable to many kinds of networks, such as roads, utilities, water, electricity, telecommunications and computer networks alike, the total number of algorithms that have been developed over the years is immense.



**TABLE OF CONTENTS**

* 1. Introduction
  2. Literature Survey
  3. Hardware & Software requirements
  4. Planning (Explanation)
  5. Implementation
  6. Results Discussion
  7. Conclusion and Future Work
  8. References

**1.Introduction**

The shortest-path problem is one of the well-studied topics in computer science, speciﬁcally in graph theory . An optimal shortest-path is one with the minimum length criteria from a source to a destination. There has been a surge of research in shortest-path algorithms due to the problem’s numerous and diverse applications. These applications include network routing protocols, route planning, traﬃc control, pathﬁnding in social networks, computer games, and transportation systems, to count a few

## Non – Adaptive Routing Algorithms

Non-adaptive Routing algorithms, also known as static routing algorithms, construct a static routing table to determine the path through which packets are to be sent. The static routing table is constructed based upon the routing information stored in the routers when the network is booted up.

The two types of non – adaptive routing algorithms are −

* **Flooding** − In flooding, when a data packet arrives at a router, it is sent to all the outgoing links except the one it has arrived on. Flooding may be uncontrolled, controlled or selective flooding.
* **Random walks** − This is a probabilistic algorithm where a data packet is sent by the router to any one of its neighbours randomly.
* In shortest path routing, the topology communication network is defined using a directed weighted graph. The nodes in the graph define switching components and the directed arcs in the graph define communication connection between switching components. Each arc has a weight that defines the cost of sharing a packet between two nodes in a specific direction.

**Literature Review**

W. Ahn and R. S. Ramakrishna [2] proposed a genetic algorithmic approach to the shortest path (SP) routing problem. Variable-length chromosomes (strings) and their genes(parameters) have been used for encoding the problem. The crossover operation exchanges partial chromosomes (partial routes) at positional independent crossing sites and the mutation operation maintains the genetic diversity of the population.

1. C. Valera and K.G. Seah [3] proposed a new routing protocol named CHAMP (Caching and Multiple Path) routing protocol. CHAMP uses cooperative packet caching and shortest multi-path routing to reduce packet loss due to frequent route failures. Extensive simulation results that these two techniques yield significant improvement in terms of packet delivery, end-to-end delay and routing overhead. It was proposed that existing protocol optimizations employed to reduce packet loss due to frequent route failures, namely local repair in AODV and packet salvaging in DSR, are not effective at high mobility rates and high network traffic.

Zwick [140] survey adopts a theoretical stand-point with regards to the exact and approximate shortest paths algorithms. Zwick’s survey addresses single-source shortest-path (SSSP), all pairs shortest-path(APSP), spanners (a weighted graph variation), and distance oracles. The survey illustrates the various variations that each category adopts when handling negative and non-negative edge weights as well as directed and undirected graphs. Sen [121] surveys approximate shortest-paths algorithms with a focus on spanners and distance oracles. Sen’s survey discusses how spanners and distance oracles algorithms are constructed and their practical applicability over a static all-pairs shortest-paths setting. Sommer [125]surveys query processing algorithms that trade-oﬀ the index size and the query time. Sommer’s survey also introduce the transportation network class of algorithms, and include algorithms for general graphs as well as planar and complex graphs. Many surveys focus on algorithms that target traﬃc applications, especially route planning methods. In such related work, a network denotes a graph. Holzer et al. [81] classify variations of Dijkstra’s algorithm according to the adopted speedup approaches. Their survey emphasizes on techniques that guarantee correctness. It argues that the eﬀectiveness of speed-up techniques highly relies on the type of data. In addition, the best speedup technique depends on the layout, memory and tolerable pre-processing time. In contrast to optimal shortest-path algorithms, Fu et al. [60] survey algorithms that target heuristic shortest-path algorithms to quickly identify the shortest-path. Heuristic algorithms aim is to minimize computation time. The survey proposes the main distinguishing features of heuristic algorithms as well as their computational costs. Goldberg [66] investigates the performance of point-to-point shortest-path algorithms over road networks from a theoretical standpoint. Goldberg reviews algorithms, e.g., Dijkstra and

 A

∗

, and illustrates heuristic techniques for computing the shortest-path given a subset of the graph. The survey proves the good worst-case and average-case bounds over a graph. Also, it discusses reach-based pruning and illustrates how all-pairs shortest-path algorithms can be altered to compute reaches while maintaining the same time bound as their original counterparts. Delling and Wagner [37]survey route planning speedup techniques over some shortest-path problems including dynamic and time-dependent variants. For example, the authors argue that shortcuts used in static networks cannot work in

**Hardware And Software Components**

**So , below ones are our Requirements with which we have implememted the project**

**Hardware and software components:**

**Laptop – Dell**

**Ram & ROM – 8GB & 512GB**

**Processor – intel core (i5 10 GEN)**

**Operating System – Windows 11**

**Software used – Cisco Packet Tracer**

**Explanation**

Consider that a network comprises of N vertices (nodes or network devices) that are connected by M edges (transmission lines). Each edge is associated with a weight, representing the physical distance or the transmission delay of the transmission line. The target of shortest path algorithms is to find a route between any pair of vertices along the edges, so the sum of weights of edges is minimum. If the edges are of equal weights, the shortest path algorithm aims to find a route having minimum number of hops

Algorithm Used:

We have used the Tree technique for Understanding purpose :

B C

A D

**2.1**

F E

Consider the diagram as a network and the labelled parts as nodes .

So, there’s a Connection between each node and each node has a function. So, A point is the source and D point is the destination. We have to find the shortest path

So, We are going to do using dijstras algorithm :

Example Involved :

For example, an individual wants to calculate the shortest distance between the source, A, and the destination, D, while calculating a subpath which is also the shortest path between its source and destination. Let’s see here how Dijkstra’s algorithm works;

It works on the fact that any subpath, let say a subpath B to D of the shortest path between vertices A and D is also the shortest path between vertices B and D, i.e., each subpath is the shortest path.

Here, Dijkstra’s algorithm uses this property in the reverse direction, that means, while determining distance, we overestimate the distance of each vertex from the starting vertex then inspect each node and its neighbours for detecting the shortest subpath to those neighbours.

This way the algorithm deploys a greedy approach by searching for the next plausible solution and expects that the end result would be the appropriate solution for the entire problem.

Implementation

Before proceeding the step by step process for implementing the algorithm, let us consider some essential characteristics of Dijkstra’s algorithm;

* Basically, the Dijkstra’s algorithm begins from the node to be selected, the source node, and it examines the entire graph to determine the shortest path among that node and all the other nodes in the graph.
* The algorithm maintains the track of the currently recognized shortest distance from each node to the source code and updates these values if it identifies another shortest path.
* Once the algorithm has determined the shortest path amid the source code to another node, the node is marked as “visited” and can be added to the path.
* This process is being continued till all the nodes in the graph have been added to the path, as this way, a path gets created that connects the source node to all the other nodes following the plausible shortest path to reach each node.

**Simulation and Testing**

In this section, we verify the improvement of our information based routing on the ability of achieving the shortest-path from a simulator, comparing with the best result. Routing is the main process to deliver packets. A router that handles a packet examines the destination address in the IP-header, computes the next hop that will bring the packet one step closer to its destination , and delivers the packet to the next hop , where the process is repeated. To make this work, two things are needed. First, routing tables match destination addresses with next hops. Second, routing protocols determine the contents of these tables. Case 1: Before starting the simulation, we first create a topology where Pc0 and Pc1 are connected with the serially connected routers(router 0 and router 1). To send a packet fromPc0 to Pc1, there is only one path. Fig. 2 and Fig. 3 show the echo packet and acknowledgement packet respectively at Pc0.The time required to reach the acknowledgement packet to Pc0 in 0.010s.Case 2: In the second topology (shown in Fig. 4) we have disconnected router0 androuter4 and connected another two routers(router1 and router3). Then we start the simulation mode. In this case, the ICMP packet is passing through router1 to reach its destination Pc1which is shown in Fig. 5. This time , the time required to send and receive ICMP packet is 0.014s (Fig. 6). Here we find that, the extra 0.004s is required because the packet has to travel more two hops than before. Case 3: We connect router0 and router4 and add a PDU again at Pc0 as shown in Fig. 7, but this time another type of packet is created. This is the ARP packet. An ARP packet is created when the host or router does not have the hardware or MAC address of its next hop. The host will send an ARP request and also drop the ICMP packet. It does not buffer the packet and wait for the ARP reply to come back because that would cause a lot of performance drop if there are a lot of pending ARP entries. When the MAC address is available on the ARP table of the router, this table can be used as next time we add PDU on the same PC. In Fig. 7, we can see that the ARP table for Pc0 is empty. After getting the ARP reply packet as shown in Fig. 8, the table is filled up with the MAC address of the next hop i.e. 192.168.1.1which is the IP address of router0.The ICMP packet is again created and forwarded. But when the packet reached to router4, it again creates an ARP packet for the same reason and gets the next hop MAC address .Case 4: We create another scenario as shown in Fig. 9. This time the ICMP packet is being passed from network to network without any abstraction (Fig. 10 and 11). The main critical point that we are trying to focus through-out these cases is the route that the packet takes to reach its destination. The packet could pass through router1

→

 router3

→

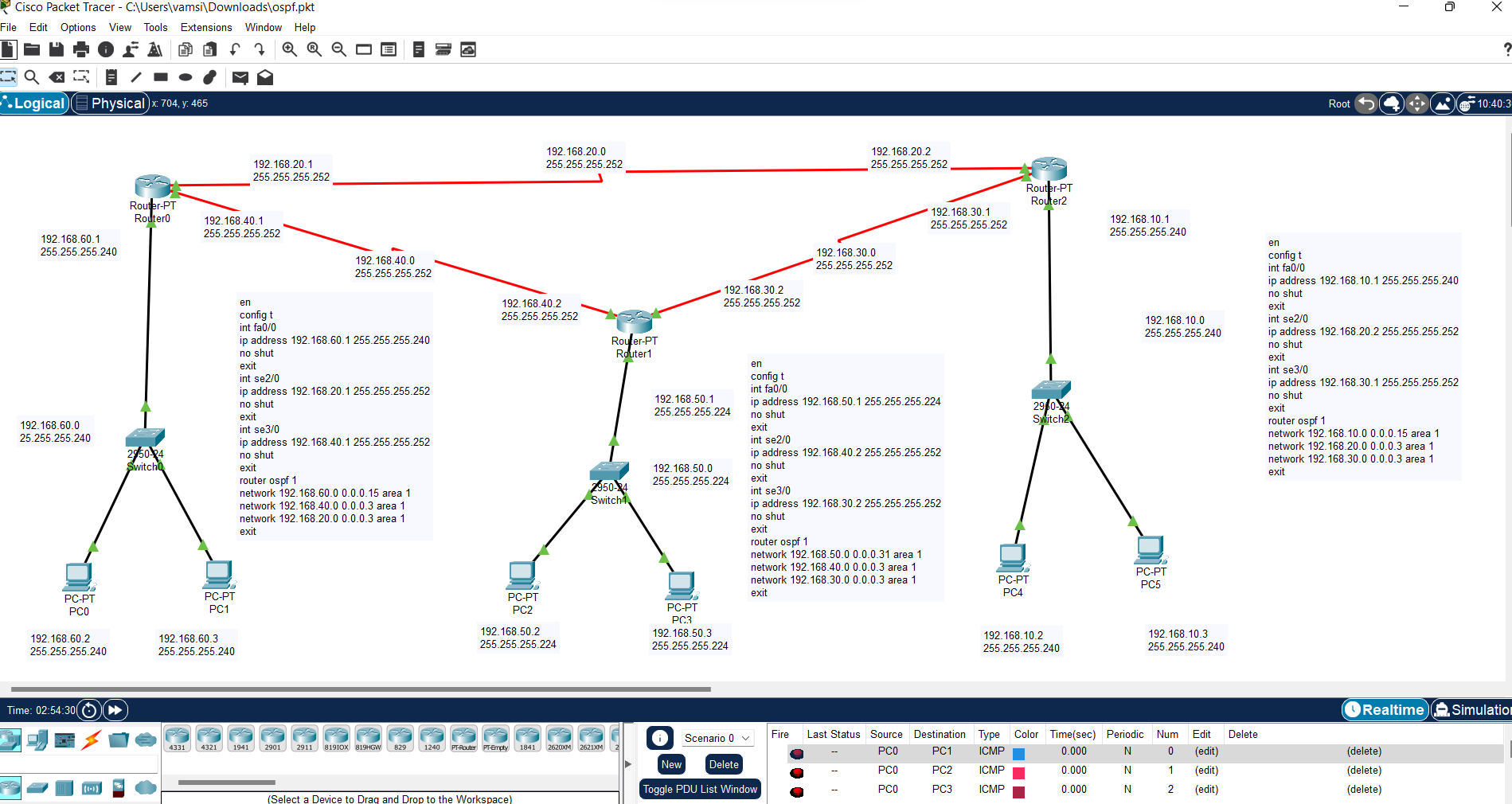
 router4; where it needs 0.014s. But it chooses its route through router0

→

router4which takes only 0.010s to complete the ICMP echo and ICMP reply. This proves the shortest path selection during transmission. Routers and switches send out CDP packets every 60 seconds regardless of what packets we have created. Each device configured

for CDP sends periodic messages, known as advertisements, to a multicast address. The advertisements also contain time-to-live, or-hold time, information, which indicates the length of time a receiving device, should hold CDP information before discarding it. Each device also listens to the periodic CDP messages sent by others in order to learn about neighbouring devices and determine when their interfaces to the media go up or down.

Let us check it using Cisco packet tracer



So , here first we have setup two PCs. Then attached that PC to that router, again done same for the

Rest of the PCs. So , Now we are giving connections through the switches using the routers.

Now we have to configure each PC along with Switches and routers . We have to configure In such a way

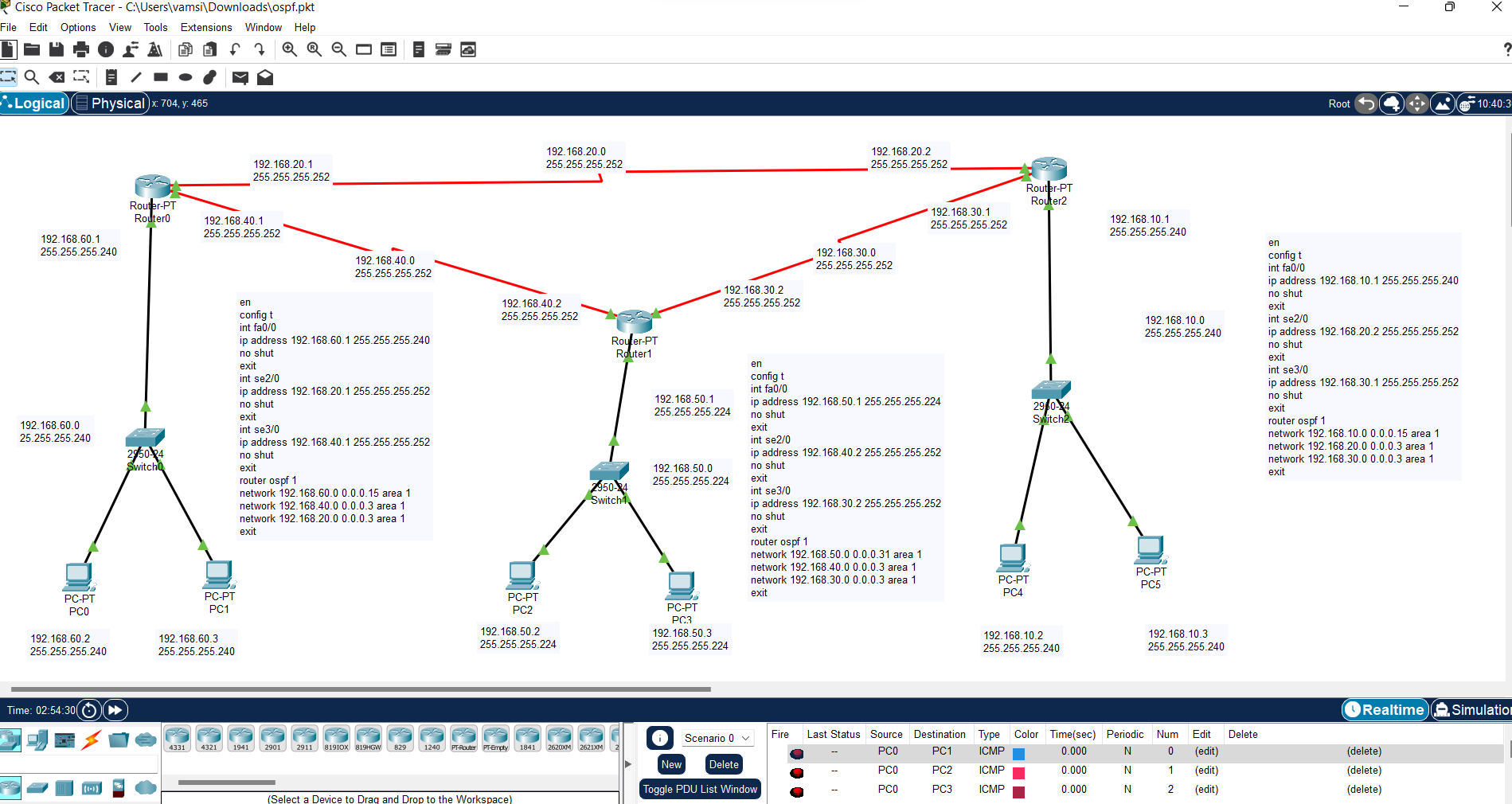
The message we send should pass in a short distance and reach to the destination. After configuring

Perfectly we have pass the message. For that we have to place a mail in one PC and other in other PC

i.e, sender mail in one PC in different switch and the receiver mail in another PC of another switch.

Although there will be more than two switches and routers we can find out the shortest path.

Result :



As we can see that the packets we have placed . The packet is transferring from the shortest path .

Conclusion :

 The analysis of shortest path in a packet racing mechanism is the common thread running through this paper. With a packet trace system we have monitored and displayed packet transferring in a telecommunication network, especially so that faults or errors in the network can be detected quickly and easily. It also allows us to find the shortest path for the router to transfer packet. In this paper, the routing protocols that have been analyzed are the formulas used by routers to determine the appropriate path onto which packet should be forwarded. The routing protocol also specifies how routers report changes and share information with the other routers in the network that they can reach. The changing conditions of the network can be dynamically adjusted by the routing protocols, otherwise all routing decisions have to be predetermined and remain static. Critical terms have been proved by using simulation software in this paper. A user interface has been used to input acceptable packet transfer characteristics. Packet transfer from network to network has been monitored at each step in the simulation which allows us to understand the trace route and find the shortest path in transmission.

References :

 Nuno M. Garcia, Przemys law Lenkiewicz , MárioM. Freire, Paulo P. Monteiro, “On the Performance of Shortest Path Routing Algorithms for Modelling and Simulation of Static Source Routed Networks – an Extension to the Dijkstra Algorithm”,

Second International Conference on Systems and Networks Communications (ICSNC2007).[2] Applied Research Group,

*IP Data Analysis;http:ipmon.sprintlabs.com/packstat/packetoverview.php*

[3] Wesam Lootah, Wil liam Enck , Patrick McDaniel, “Ticket-based Address Resolution protocol”, 21

st

 Annual Computer Security Applications Conference (ACSAC’05), Issue December 2005, pp. 106-116.[4] D. C. Plummer. An ethernet address resolution protocol or converting network protocol addresses to 48.bit ethernet address for transmission on ethernet hardware. RFC 826, November 1982.[5] D. Bruschi, A. Orgnaghi, and E. Rosti. S-arp: asecure address resolution protocol. 2003