REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Information Technologies

A NOVEL ALGORITHM FOR SELECTING OPTIMAL ROUTING PATHS TO IMPROVE NETWORK PERFORMANCE

Mohammed Salah Aldeen Ahmed AHMED

Master's Thesis

Supervisor

Asst. Prof. Dr. Oğuz KARAN

Istanbul, 2023

A NOVEL ALGORITHM FOR SELECTING OPTIMAL ROUTING PATHS TO IMPROVE NETWORK PERFORMANCE

Mohammed Salah Aldeen Ahmed AHMED

Information Technologies

Master's Thesis

ALTINBAŞ UNIVERSITY

2023

This thesis title A NOVEL ALGORITHM FOR SELECTING OPTIMAL ROUTING PATHS TO IMPROVE NETWORK PERFORMANCE prepared by MOHAMMED SALAH ALDEEN AHMED AHMED and submitted on 11/04/2023 has been **accepted unanimously** for the degree of Master of Science in Information Technologies.

____
Asst. Prof. Dr. Oğuz KARAN
Supervisor

Thesis Defence Committee Members:

Asst. Prof. Dr. Oğuz KARAN	Department of Software Engineering, Altınbaş University	_______________
Asst. Prof. Dr. Sefer KURNAZ	Department of Computer Engineering, Altınbaş University	_______________
Asst. Prof. Dr. Serdar KARGIN	Department of Biomedical Engineering, Arel University	_______________

I hereby declare that this thesis meets all format and submission requirements of a Master's thesis.

Submission date of the thesis to Institute of Graduate Studies: ___/___/___

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Mohammed Salah Aldeen Ahmed AHMED

Signature

DEDICATION

First, I would like to thank Allah Almighty for the power of the mind, health, strength, guidance, knowledge, and skills to complete this study, This thesis is wholeheartedly dedicated to my parents. There are no words to describe what you, mean to me, there is nothing that I can repay for what you have done to me. I will continue to do my best to achieve your expectations Lastly, I dedicated this to the family, relatives, and friends who have been encouraging me during this study.

ABSTRACT

A NOVEL ALGORITHM FOR SELECTING OPTIMAL ROUTING PATHS TO IMPROVE NETWORK PERFORMANCE

AHMED, Mohammed Salah Aldeen Ahmed

M.Sc., Information Technologies, Altınbaş University,

Supervisor: Asst. Prof. Dr. Oğuz KARAN

Date: 04/2023

Pages: 64

In this research, we explore various path methods utilizing Multiprotocol Label Switching (MPLS) to improve the term Traffic Engineering (TE) in networks. As internet services continue to grow in importance, efficient routing algorithms are crucial for TE. MPLS is a contemporary technique that improves router speed, identifies explicit routers, and facilitates better Traffic Engineering Management and real-time multimedia data communication. Among the protocols utilized to determine the shortest path for packet transmission in a network, MPLS is a significant one. However, ISPs must enhance the routing algorithm on MPLS routers to optimize their efficiency. Advanced routing algorithms have been developed to leverage MPLS networks, utilizing traffic-based routing to find label switching path (LSP) solutions and reduce blocking probability. To evaluate these algorithms, we utilized Matlab to simulate the MPLS network. The simulation results were analyzed and evaluated from various perspectives, and recommendations were made for specific routing algorithms based on the network specification protocol.

The outcomes of this research indicate that integrating path methods for computer networks in terms of packet switching provides the best optimal route path.

Keywords: Path, MPLS, Network, End to End Delay, Anti Colony Method.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

CSR	:	Cell Switching Router
MPLS	:	Multiprotocol Label Switching
LER	:	Label Edge Router
LMP	:	Ling Management Protocol
IGP	:	Internal Gateway Protocol
IDS	:	Instruction Detection System
SIDS	:	Signature-based IDSs
TE	:	Technical Engineering
NIDS	:	Network Intrusion Detection System
DIDS	:	Distributed Intrusion Detection System

1. INTRODUCTION

1.1 BACKGROUND

In the past two decades, various network protocols have been developed by scientists and companies worldwide to build internet networks and other types of networks. These protocols have made networks more comprehensive, especially in the industrial field, where new protocols are continuously being tested and implemented [1]. Previously, the main objective of networks was to connect source and destination nodes through IP assignments. However, this method was unreliable and had significant time delays, causing many problems [2]. Hence, finding an intelligent algorithm to determine the best path between source and destination nodes has become crucial. One solution to overcome issues caused by disasters or unexpected errors is to dynamically change the IP, which can help the system handle various problems, especially in large-scale systems. Today's network must meet the demands of modern applications, to cater to the demands of online shopping and banking services, it is essential to provide functionalities like real-time processing, fast speed, and strong security measures. [3]. Next-generation systems need to support real-time applications, voice and video applications, and services like IP Telephone, which require the shortest path between two nodes to be predicted and found. The use of Virtual Private Networks (VPNs) encourages IP technology providers to explore supplementary techniques for efficient traffic management within IP protocols. Therefore, internet service providers need to provide high-performance services and efficiently find the best path. In MPLS, packets are labeled with specific identifiers that enable routers to direct them along a predetermined path, rather than individually determining the best route for each packet. This allows for faster and more efficient data transfer across the network. Furthermore, MPLS enables the creation of Virtual Private Networks (VPNs), which provide secure communication channels between geographically dispersed sites [5]. By using MPLS, organizations can reduce costs associated with dedicated lines or leased circuits while maintaining a high level of security. MPLS also offers Quality of Service (QoS) features, allowing for prioritization of traffic based on its importance and ensuring that critical data is transmitted without delay or loss. Overall, the adoption of MPLS has significantly improved network performance, reduced costs, and provided organizations with a more reliable and secure means of communication [6]. The subsequent are the main challenges confronted by networks universally:

a. Expandability: Networks must be competent of increasing in scope without any virtual restraints. Data must grow speedily, and the network must be capable of accommodating this growth each day. All of this should be attainable on the router at a minimum.

b. Adaptability: The capability to transform and progress without interrupting network distribution is one of the most noteworthy challenges.

c. Consolidation: The incorporation of numerous applications is a manifestation of system consolidation. The weight of consolidation at all strata cannot be overemphasized in generating an interactive network [7].

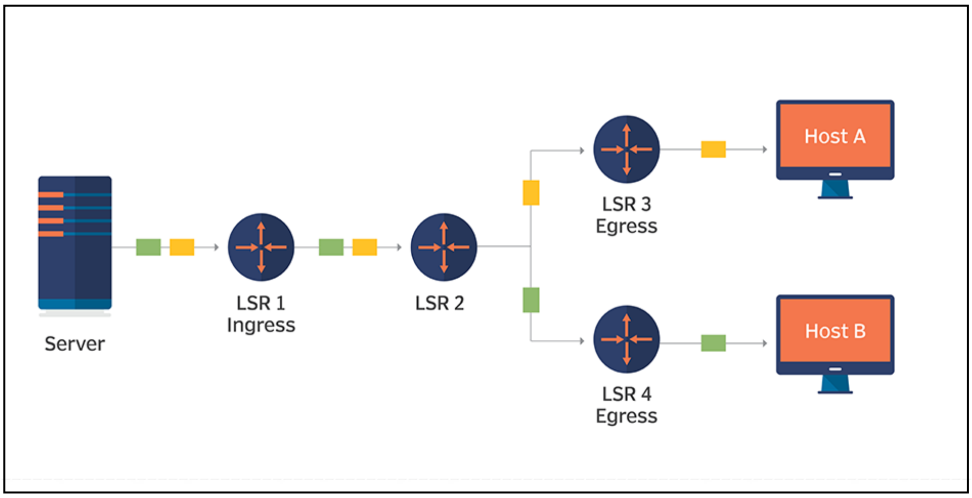

Figure 1.1: Concept Of MPLS [6].

1.2 RELATED WORK

In the context of MPLS networks, many studies have investigated routing algorithms and models. These studies have focused on various aspects, such as scalability, fault tolerance, traffic engineering, and energy efficiency.

the study by [8], proposed an optimized routing algorithm that considers network congestion and link failures. Their proposed algorithm achieved better performance than existing algorithms in terms of network throughput, packet loss rate, and energy consumption.

Another study by [9] proposed a hierarchical routing model that reduces the control overhead and enhances the scalability of MPLS networks. They showed that their proposed model can achieve better performance than existing models in terms of routing convergence time and control message overhead.

In a different study, [10], proposed a path selection algorithm that considers both traffic engineering and fault tolerance. Their algorithm aims to minimize network congestion while maintaining high availability in the event of link failures. Their simulation results showed that their algorithm achieved better performance than existing algorithms in terms of network throughput, packet loss rate, and recovery time.

Other studies have focused on different aspects of MPLS networks, such as energy efficiency [11] network security [12], and dynamic routing [13].

Using sensors and network mapping, the authors in [14] proposed techniques to perform experiments in collecting and analyzing the data collection, and analyzing traffic behavior, and data analysis. Furthermore, to visualize network data, the techniques provided several ways. Moreover, the techniques dealt with identifying the application. The authors in [15] explained the need for network proxying internet system and presented the network monitor's structure and operations. This article inspired us to conduct this research because it contained the network security monitoring aspects.

The authors in [36] utilized the KDD dataset and used several filter algorithms to reduce forty-one features to sixteen one to achieve the same detection outcomes. They utilized Stability Selection, Stepwise Regression, and Maximum Relevance (WMR). The author validated their work with the use of Bayes Classifier and Support Vector Machines (SVM). We were inspired to inquire about the necessity of extremely complex extracted characteristics by the study. Furthermore, a more basic feature set is key to performing the current research tests.

Overall, these studies demonstrate the importance of routing algorithms and models in the performance of MPLS networks. They also highlight the need for continued research in this field to address emerging challenges and improve the resilience and reliability of MPLS networks.

Table 1.1: Review Of Related Area Research Of Network.

Year	Published	Paper details	Demerits
2022	[14]	SDN technology has various applications, including routing, monitoring, and security. It can also enhance cellular and satellite networks, optimize workload placement in data centers, and benefit from evaluation tools like Mininet and OpenNet. However, there are future challenges that need to be addressed in the implementation and adoption of SDN technology.	The focus is solely on routing methods that utilize the principles of software-defined networking.
	[15]	The paper is being addressed the classification and applications of unmanned aerial vehicles (UAVs), as well as the different mobility models and routing protocol classifications that are relevant to these devices. Additionally, the challenges and open issues that still need to be resolved in this field are also being discussed.	The analysis of predictive techniques is not comprehensive, and it does not encompass routing protocols that are based on self-adaptive learning. These types of routing protocols are not included in the review.

Table 1.1: Review Of Related Area Research Of Network "Tables Continued".

[16]	The discussion pertains to various aspects of communication networks for unmanned aerial vehicles (UAVs), such as the issues, characteristics, design considerations, and applications. It also covers the different routing protocols used in these networks, as well as the quality of service (QoS) aspects that need to be considered. Lastly, the future research areas that require further exploration in this field are also being examined	Not included path routing protocols
[17]	The topic being addressed includes the architecture and design of networks, as well as the different routing protocols that are utilized in these networks. Additionally, performance analysis and quality of service (QoS) metrics are also being discussed, along with the current opening issues that require further attention in this field.	The routing classification being discussed is limited to only a few methods.

Table 1.1: Review Of Related Area Research Of Network "Tables Continued".

[18]	The paper under consideration explores the various ways in which unmanned aerial vehicles (UAVs) can be utilized for data gathering and monitoring, cellular communications, disaster management, computing, and routing methods in conjunction with unmanned ground vehicles (UGVs).	Does not include any routing protocols for communication between UAVs.
[19]	The topic being discussed pertains to routing protocols that are specifically designed for network of UAV-assisted VAN. Additionally, the analysis covers the current challenges and open research issues that need to be addressed in this field..	The analysis or discussion is solely focused on routing protocols that are designed for UAV-aided vehicular networks
[20]	The topic being discussed pertains to various aspects of unmanned aerial vehicle (UAV).	There is a lack of discussion on routing protocols that incorporate artificial intelligence and how they can affect networks that are constantly changing or in motion.
[14]	The analysis encompasses a range of subjects, including network design, mobility models, routing approaches, and protocols. Furthermore, it features a comparative analysis of these topics, as well as a discussion of future challenges that must be tackled in this domain.	There is no consideration given to self-adaptive learning-based methods.

Table 1.1: Review Of Related Area Research Of Network "Tables Continued".

[15]	The proposed approach for path restoration involves utilizing an Integer Programming method. The method is characterized by limited fixed routes and includes joint path computation..	The discussion provides only a brief mention of adaptive routing protocols, but there is a notable absence of AI-enabled routing protocols..
[16]	The analysis pertains to a comparison of different routing protocols, as well as the open research issues that need to be addressed in this field.	There is a lack of discussion on routing protocols that incorporate artificial intelligence and how they can affect networks that are constantly changing or in motion.
[17]	The paper being discussed is related to both single-layer and cross-layer routing techniques, as well as the challenges and future research directions that need to be explored in this area.	The routing classification being discussed is limited to only a few methods., but not defined for UAV Networks.
[18]	The discussion covers various aspects of UAVs, including their architectures, ongoing projects, distinguishing characteristics, potential applications, and routing protocols. Additionally, there is a particular emphasis on the security challenges that UAVs present.	Not covered the routing path in terms of AI

Table 1.1: Review Of Related Area Research Of Network "Tables Continued".

[19]	The examination centers on routing protocols based on position and presents an intricate explanation and comparative investigation of these methods. Moreover, the conversation also encompasses several mobility models and the use of UAVs in various applications within this domain.	The discussion is only, limited to position-based routing protocols,
Our Propose work	Focused on MLSP in computer network networks, which involves a comparison of various methods to determine the optimal route.	Find that integrating protocol method is the best method for the optimal route method.

1.3 PROBLEM STATEMENT

Packet switching networks rely on recovery mechanisms to maintain network connectivity in the event of failures. However, these mechanisms can lead to decreased throughput and increased loss rate. Recovery success depends on routing protocols that may not guarantee the same success rate for all failures. This research aims to optimize the throughput of packet switching networks by identifying ways to reduce loss rate and increase connection rates. It will investigate the impact of network size and routing protocols on the recovery process and propose strategies to enhance recovery performance. Ultimately, the goal is to develop a comprehensive approach to improve the resilience and reliability of packet switching networks.

1.4 RESEARCH OBJECTIVE

The objective of this study is to analyze the performance of three algorithms commonly used in software defined networking (SDN) networks in achieving resilient traffic flow conditions in case of network failures. Specifically, the study aims to explore and compare the effectiveness of different routing techniques in ensuring continuous traffic flow when a network fails. The three algorithms under analysis are expected to provide insights into how networks can be designed to improve their resiliency and minimize downtime in the event of network failures.

Overall, the study seeks to contribute to the advancement of SDN networks and inform network designers on the best practices for ensuring reliable and robust network performance.

1.5 THESIS OUTLINES

This thesis is focused on the analysis of different techniques that can be used for routing traffic in the event of network failure. The main objective is to investigate the performance of three algorithms commonly used in the design of networks, particularly in software-defined networking (SDN) networks, and their ability to achieve resilient traffic flow conditions in such an environment. Chapter 1, the first chapter of the thesis will provide an overview of the research topic, including a background on the importance of resilient traffic flow in networks. It will also introduce the concept of SDN and its role in traffic routing, as well as identifying the research gap in the current literature. The chapter will outline the research questions, objectives, and significance of the study. Chapter 2 will provide a comprehensive explanation of the theoretical background of the topic. This will include a detailed description of SDN and its advantages over traditional networking, an explanation of traffic routing in SDN networks, and an introduction to the three algorithms under analysis. The chapter will also discuss resiliency in network design and its importance in improving network performance. Chapter 3 will outline the methodology of the research, including research design and methodology, data collection methods, data analysis techniques, and proposed methods for evaluating the three algorithms. The chapter will also address the limitations of the study and how they will be addressed. Chapter 4 results and Analysis Chapter 4 will present the results of the study and compare the performance of the three algorithms under analysis. It will provide findings from data analysis, a comparison of the algorithms in terms of their ability to ensure resilient traffic flow, and discussion of the implications of the results for network design. The chapter will also provide recommendations for the best algorithm to use in different scenarios. Chapter 5 Conclusion and Future Work, the final chapter of the thesis will summarize the findings of the study and discuss their significance for future research and practice. It will provide a summary of the research questions and objectives, the findings, and how they address the research gap. The chapter will also discuss limitations of the study and recommendations for future research, conclusions, and implications for network design. The importance of resiliency in network design and its potential to improve network performance will also be emphasized.

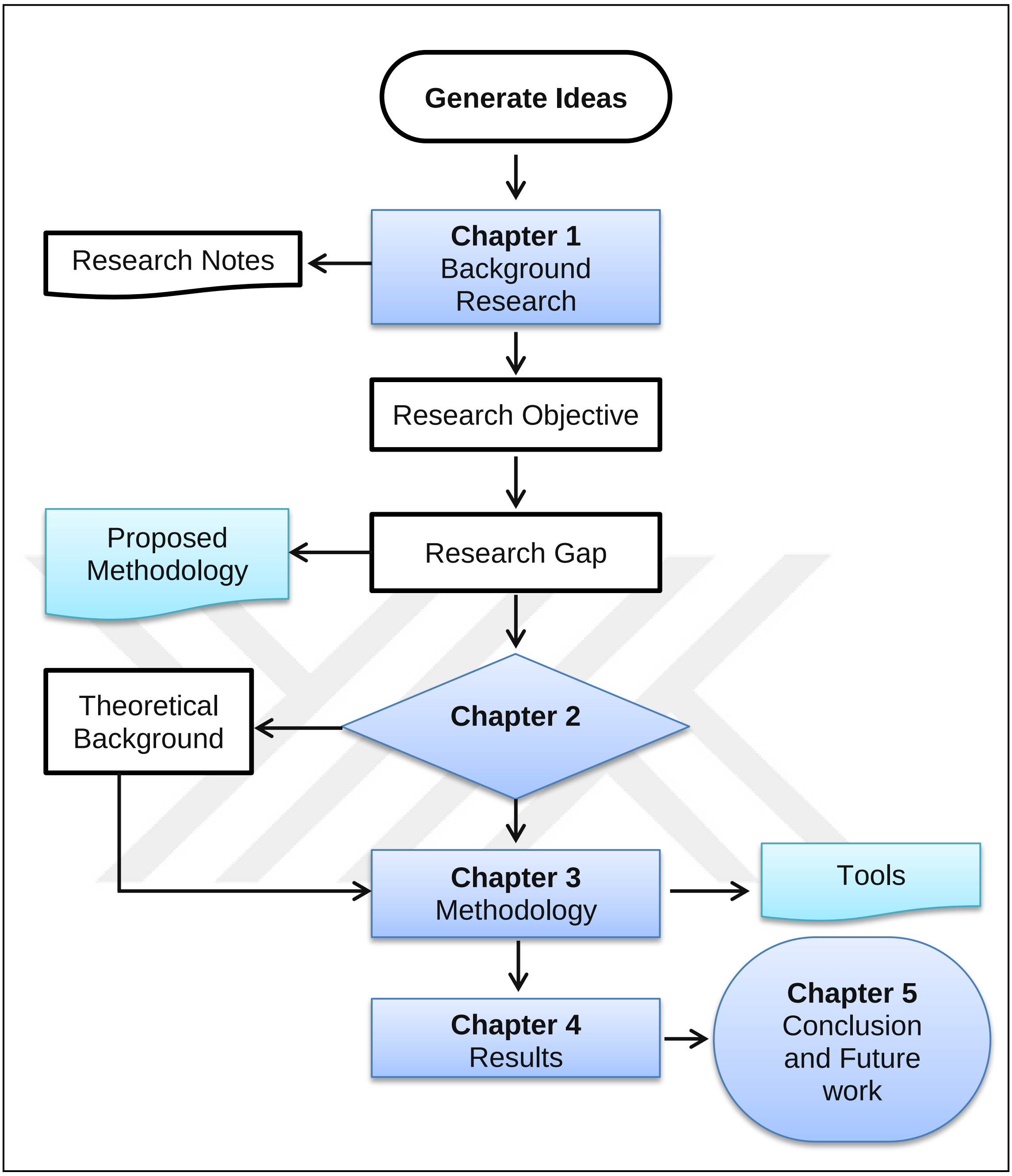

Figure 1.2: Flowchart Shows The Outline Of The Thesis

2. THEORETICAL BACKGROUND

2.1 NETWORK FUNCTIONAL REQUIREMENT

To obtain the most comprehensive view of network activity, the feature collector should be able to parse a broad range of the protocols. The system should also report traffic that it cannot comprehend.

2.1.1 Data Availability

Prior to creating summary structures, network monitor must make collected data accessible at minimal level that is possible for the research purposes. This means that for each protocol that the monitor could interpret, protocol-specific information must be provided [20].

2.1.2 Abstraction Capabilities

Summary structures, abstractions, aggregate values, and statistical data could be pro-duced from acquired data with the use of information which was acquired during the collecting stage. Bidirectional connections and unidirectional network flows are common examples of summary structures [21].

2.2 OVERVIEW OF MPLS

Multiprotocol Label Switching (MPLS) is a technique used in computer networks to direct data packets along predetermined paths, called Label Switched Paths (LSPs), based on labels that are attached to each packet. These labels are used to identify the next hop in the path and to enable fast forwarding of packets through network routers [22]. As shown in the figure 2.1 indicates the control panel of MPLS.

The basic operation of MPLS can be described mathematically as follows:

a. Label Distribution Protocol (LDP): The first step in setting up an LSP is to establish label mappings between the network routers along the path. This is done using a protocol called the Label Distribution Protocol (LDP), which assigns a unique label to each network prefix or route. Let's assume that there are n routers along the path, each with a unique IP address [23].

b. Label Stack: When a data packet arrives at a router, it is assigned a label by the router based on the LDP mappings. This label is added to the top of the packet as a label stack. Let's assume that the label assigned to the packet by the first router is L1 [24].

c. Label Switching: The packet is then forwarded to the next hop in the path, based on the topmost label in the label stack. The next router examines the label and uses it to determine the next hop in the path. The label is then popped off the stack, and the packet is forwarded to the next hop based on the new topmost label. This process continues until the packet reaches its destination [24].

d. Label Swapping: In some cases, a router may need to swap the label on a packet, for example, if it needs to redirect the packet to a different path. In this case, the router replaces the topmost label with a new label, based on the LDP mappings for the new path [22].

e. Path Setup: When a new LSP is established, the routers along the path exchange information to set up the path and determine the labels to be used. This involves sending messages to each other to establish the LDP mappings, and to negotiate the path and label assignments [25].

Furthermore, MPLS is a technique that enables fast forwarding of packets through a network by assigning labels to packets and using these labels to direct packets along predetermined paths. The label mappings are established using the Label Distribution Protocol, and the labels are added to the packet as a label stack. The routers along the path use the labels to determine the next hop in the path and to forward the packet to the appropriate destination.

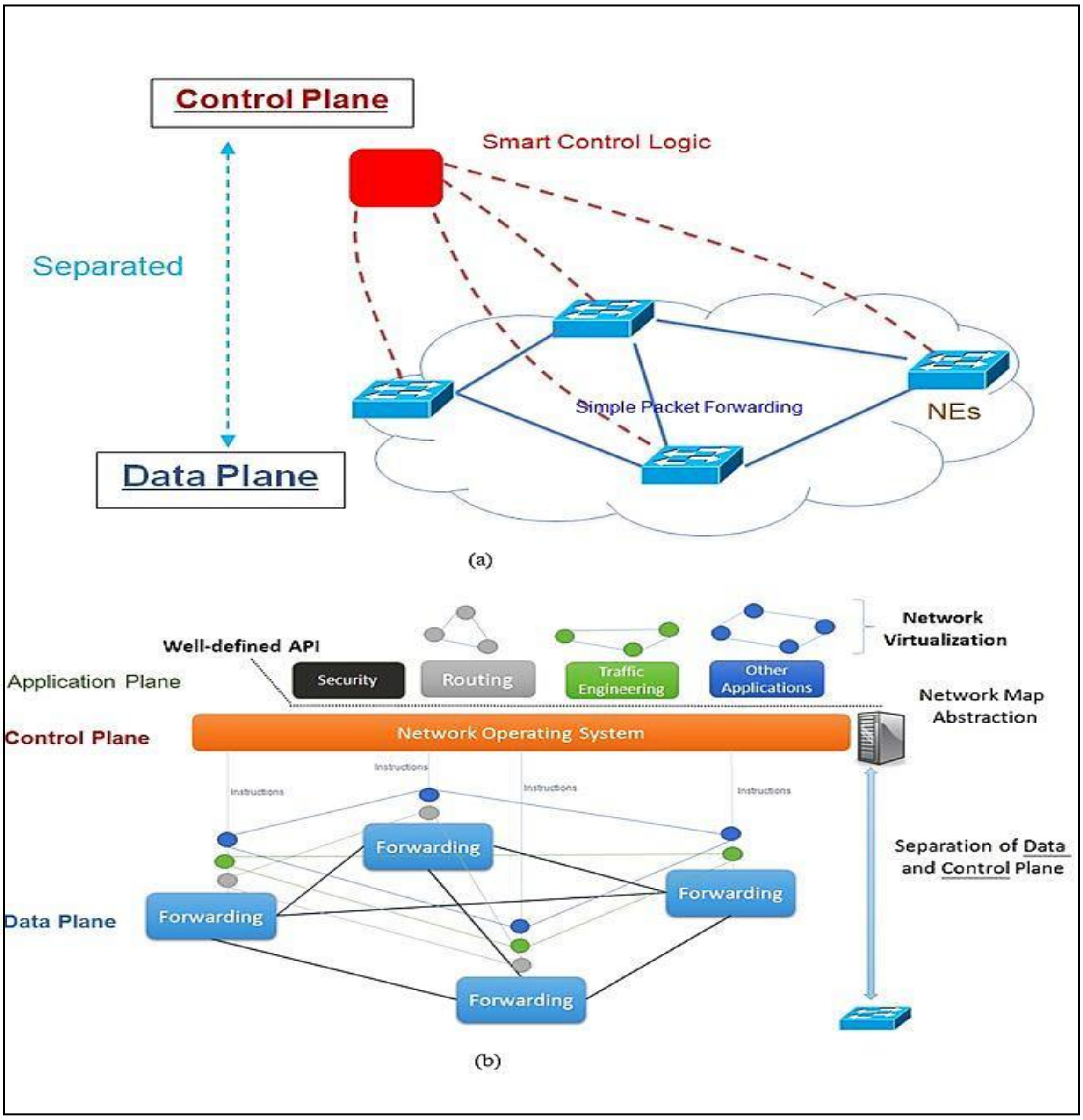

Figure 2.1: (a) Simple Packet Forwarding (b) Application Plane [26].

2.3 TRAFFIC ENGINEERING

in terms of reliability, efficiency, and speed. By allocating bandwidth in a more efficient and optimized way, network engineers can ensure that critical services are given priority and that there is sufficient capacity to handle peak traffic loads. Traffic engineering involves analyzing the network's current traffic patterns and identifying potential bottlenecks or areas of congestion. Based on this analysis, network engineers can adjust the routing paths and allocate bandwidth in a way that optimizes network performance [27]. This can involve adjusting link weights, setting up Quality of Service (QoS) policies, and implementing traffic shaping or policing mechanisms to ensure that certain types of traffic are given priority [28].

Overall, the goal of traffic engineering is to ensure that the network is able to handle current and future traffic loads while maintaining high levels of performance and reliability. By using tools and techniques to monitor and optimize traffic flow, network engineers can ensure that their network is operating at its best and delivering the best possible user experience.

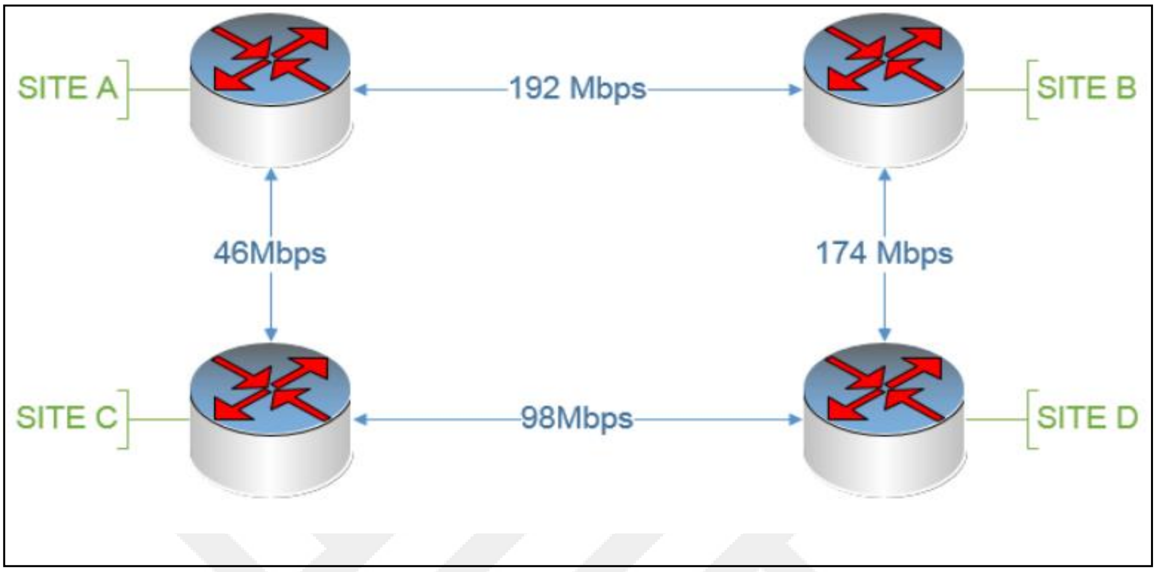

Figure 2.2: Traffic Engineering With IP Process [28].

All of the routers in a site are equipped with an IGP configuration, which allows them to advertise their networks between them. Figure 2.2, shows a network composed of four links with varying capacities. If one wants to communicate with another site, it can do so through site B or through site C. A router will dynamically select the link with the highest capacity to send data unless a policy is explicitly set in place.

2.4 IP ROUTING FUNCTIONS

The routing table depicted in Figure 2.3 guides the IP network router in determining the next destination for packets. This process is repeated by each router in the path until the packet reaches its intended destination [29].

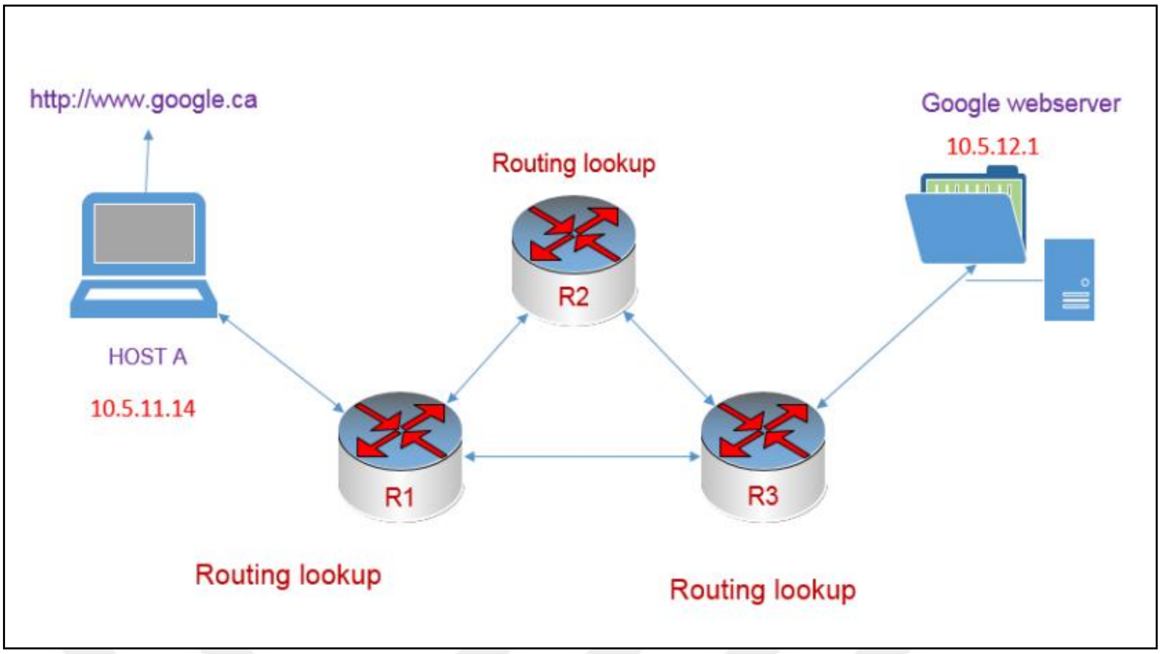

Figure 2.3: Routing Lookup In Network [30].

The sites contain networks that have an IGP configuration. Each site has four links as shown in Figure 2.3, for link capacity. To communicate with another site, one can use site B or site C, and by default, a router will dynamically route traffic between all sites. If a host with an IP address of 10.5.11.4 needs to access the web server, it will send a packet to the appropriate router on the path to the destination, and the router will keep the destination in its routing table. This information helps the router to determine the path that the packet will take to reach its destination. Local networks use the OSPF database to track all connections, which can also be used for larger networks. The Internet has autonomous systems consisting of routers that are under one administrator's control.

2.5 AUTONOMOUS SYSTEM

An Autonomous System (AS) is a collection of interconnected networks that are operated by a single administrative entity, such as an internet service provider (ISP), a large organization, or a government agency. Each AS is assigned a unique identifier known as an Autonomous System Number (ASN), which is used to exchange routing information with other ASs on the Internet. ASs are typically classified into two categories: transit ASs and stub ASs. Transit ASs provide connectivity between other ASs, while stub ASs only have a single connection to another AS and do not provide transit services [31].

The mathematical basis of AS routing is based on the Border Gateway Protocol (BGP), which is used by routers in different ASs to exchange routing information. BGP uses a path-vector algorithm to determine the best path for routing packets between ASs. Each AS advertises the networks it can reach to its neighboring ASs, along with a list of ASs that packets should traverse to reach those networks. BGP also employs a number of policies that ASs can use to control the flow of traffic through their networks. For example, an AS may use a policy called "prepending" to make its network appear less desirable to other ASs. This is done by adding its own AS number to the list of ASs that packets must traverse to reach a destination network. By doing so, the AS makes it appear as if there are more hops required to reach the destination, which may cause other ASs to prefer a different path. Another common BGP policy is called "route filtering," which is used to control which networks are advertised to neighboring ASs. ASs may use route filtering to prevent traffic from flowing through their networks to certain destinations, or to prioritize traffic to certain networks over others [32].

Overall, AS routing is a complex and dynamic process that is essential for the functioning of the Internet. The math behind AS routing is based on the path-vector algorithm used by BGP, along with various policies and techniques that ASs can use to control the flow of traffic through their networks.

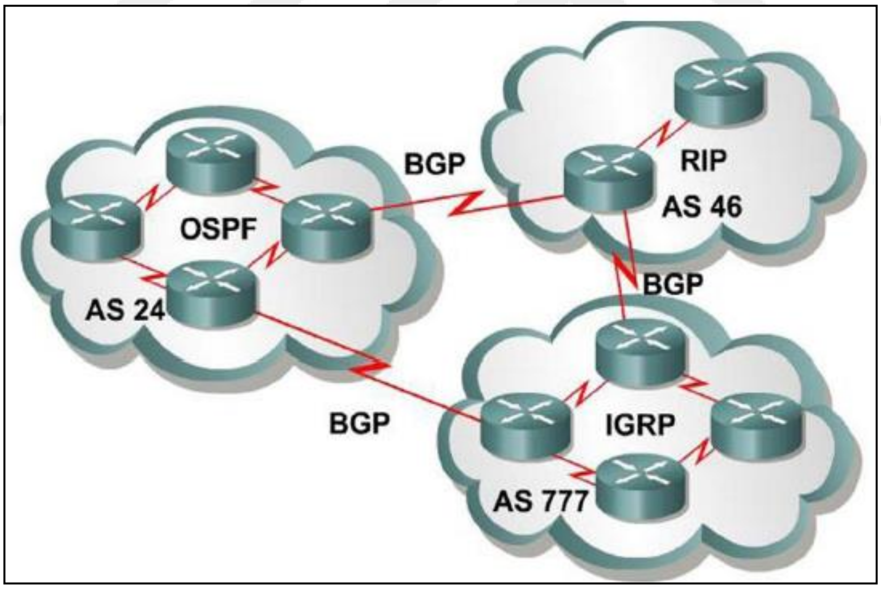

Figure 2.4: Network Represents AB System [32].

2.6 NETWORK ROUTE

Network routing refers to the process of determining the path that data packets should take through a network to reach their destination. This process is critical for the efficient and reliable transfer of data between devices on a network, and it is typically implemented using a routing protocol [33]. A routing protocol is a set of rules and procedures that govern how routers communicate with each other to determine the best path for routing packets. There are several routing protocols in use today, including the Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Border Gateway Protocol (BGP). The basic principle of network routing is to find the shortest path between the source and destination devices, which minimizes the number of routers that the data must pass through and reduces the latency or delay in transmitting the data. To achieve this, routing protocols use various algorithms to calculate the optimal path for routing packets. One such algorithm is the Dijkstra algorithm, which is used in OSPF and other link-state routing protocols. The Dijkstra algorithm works by constructing a network topology graph, which represents the network as a set of nodes and edges. Each node represents a router, while each edge represents a link between two routers. The algorithm then computes the shortest path from the source node to all other nodes in the network, taking into account the weights or costs associated with each edge [34].

Another algorithm used in routing is the Bellman-Ford algorithm, which is used in distance-vector routing protocols like RIP. The Bellman-Ford algorithm works by having each router broadcast its entire routing table to its neighbors, which in turn broadcast their own routing tables. The routers then use this information to calculate the shortest path to each destination network, based on the number of hops or distance required to reach it. In addition to determining the optimal path for routing packets, routing protocols also implement various techniques to ensure reliability and fault tolerance in the network [35]. For example, routing protocols may use redundancy and load balancing to ensure that there are multiple paths available for routing data, in case one path fails or becomes congested. Overall, network routing is a complex process that involves various algorithms and techniques for determining the best path for routing packets through a network. The selection of a routing protocol depends on the specific requirements and characteristics of the network, such as its size, topology, and traffic patterns [36].

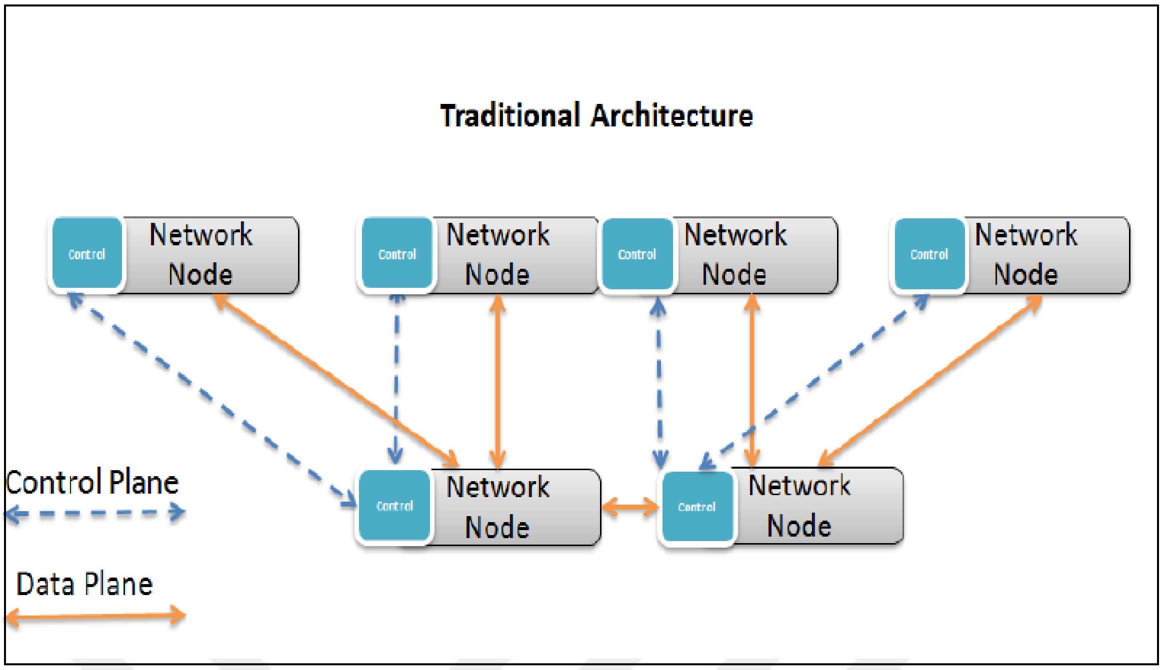

Figure 2.5: Network Route Process [35].

2.7 ROUTING INFORMATION PROTOCOL

The RIP (Routing Information Protocol) is a distance-vector routing protocol used for managing and exchanging routing information between routers in a network. The RIP protocol was originally developed for use in small to medium-sized networks and is now widely used in both LAN and WAN environments [36]. In this response, I will provide a detailed explanation of the RIP protocol, including its features, advantages, and limitations. As shown in the figure 2.6 how the RIP works.

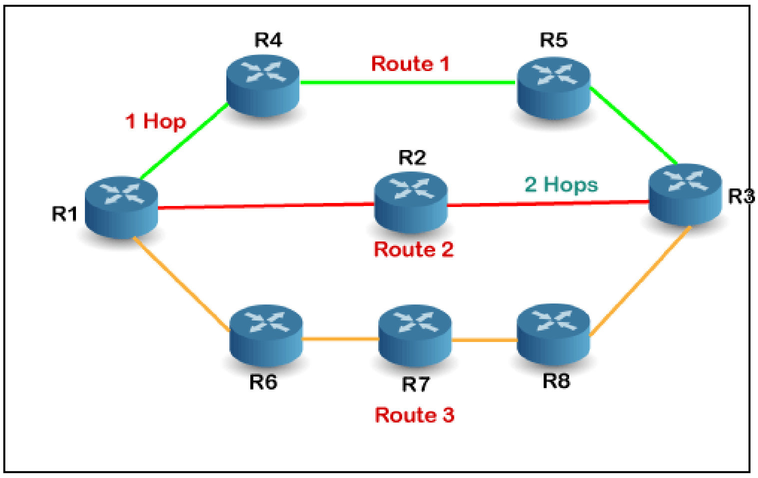

Figure 2.6: RIP Steps Of Working [36].

2.7.1 Features Of The RIP Protocol

The RIP protocol has several key features that make it useful for managing and exchanging routing information between routers [37]. These include:

a. Distance-vector protocol: The RIP protocol is a distance-vector protocol, which means that it uses the number of hops or distance between routers as a metric for selecting the best path for data packets. Each router maintains a routing table that contains information about the available paths and their associated metrics.

b. Hop count metric: The RIP protocol uses the hop count metric to determine the best path between routers. A hop count is the number of routers that a data packet must pass through before reaching its destination. The RIP protocol selects the path with the lowest hop count as the best path.

c. Periodic updates: The RIP protocol uses periodic updates to exchange routing information between routers. Routers exchange routing information every 30 seconds by default. If a router does not receive updates from its neighbor within 180 seconds, it assumes that the neighbor is no longer reachable.

d. Routing table: Each router maintains a routing table that contains information about the network topology, including the IP addresses of all routers in the network, the hop count to each router, and the outgoing interface for each path.

e. Convergence time: The RIP protocol has a relatively fast convergence time, which means that the time taken for routers to adjust their routing tables in response to changes in the network topology is relatively low.

2.7.2 Advantages Of The RIP Protocol

The RIP protocol has several advantages that make it a popular choice for small to medium-sized networks [38]. These include:

a. Simplicity: The RIP protocol is relatively simple to configure and manage, making it a popular choice for small networks with limited resources.

b. Low bandwidth requirements: The RIP protocol uses periodic updates to exchange routing information, which reduces the amount of bandwidth used compared to other protocols that use continuous updates.

c. Convergence time: The RIP protocol has a relatively fast convergence time, which means that routers can quickly adjust their routing tables in response to changes in the network topology.

2.7.3 Limitations Of The RIP Protocol

The RIP protocol also has several limitations that should be considered when choosing a routing protocol for a network [38]. These include:

a. Scalability: The RIP protocol is not well-suited for large networks with many routers. As the network size increases, the amount of time required for routers to exchange routing information increases, which can lead to slow convergence times and network instability.

b. Limited hop count: The RIP protocol has a maximum hop count of 15, which means that it cannot be used for networks with more than 15 hops between routers.

c. Slow convergence time for large networks: As the network size increases, the time taken for routers to adjust their routing tables in response to changes in the network topology increases, which can lead to slow convergence times and network instability.

d. Vulnerable to routing loops: The RIP protocol is vulnerable to routing loops, which can occur when two or more routers have incorrect routing information. This can cause data packets to be sent in circles between routers, leading to network congestion and instability.

In summary, the RIP protocol is a distance-vector routing protocol used for managing and exchanging routing information between routers in a network. The RIP protocol is simple to configure and manage, has low bandwidth requirements, and has a relatively fast convergence time. However, it is not well-suited for large networks and is vulnerable.

2.8 ROUTE STRUCTURE TABLE

A route table is a key component of a computer network that is used to determine how packets of data are forwarded from one network location to another. The structure of a route table typically consists of several columns that contain information about the available network paths and their associated characteristics [39]. The exact structure can vary depending on the specific routing protocol being used and the requirements of the network. Figure 2.7 indicts the route table of the router in the computer network.

Some of the common fields that may be found in a route table include:

a. Destination network - this field specifies the network or subnet that the route is intended for.

b. Subnet mask - this field indicates the size of the network, and is used to determine which IP addresses are part of the network and which are not.

c. Next hop - this field identifies the next device or network interface that should be used to forward packets towards the destination network.

d. Interface - this field identifies the local network interface that packets should be sent out on.

e. Metric - this field provides a measure of the distance or cost associated with a particular route, and is used to determine the optimal path for forwarding packets.

f. Flags - this field contains various flags that are used to indicate special routing conditions, such as whether a route is active or inactive.

In addition to these fields, a route table may also contain other information such as administrative distance, which is used to resolve conflicts between routes from different sources, and route summarization information, which is used to reduce the size of the routing table by consolidating multiple subnets into a single entry.

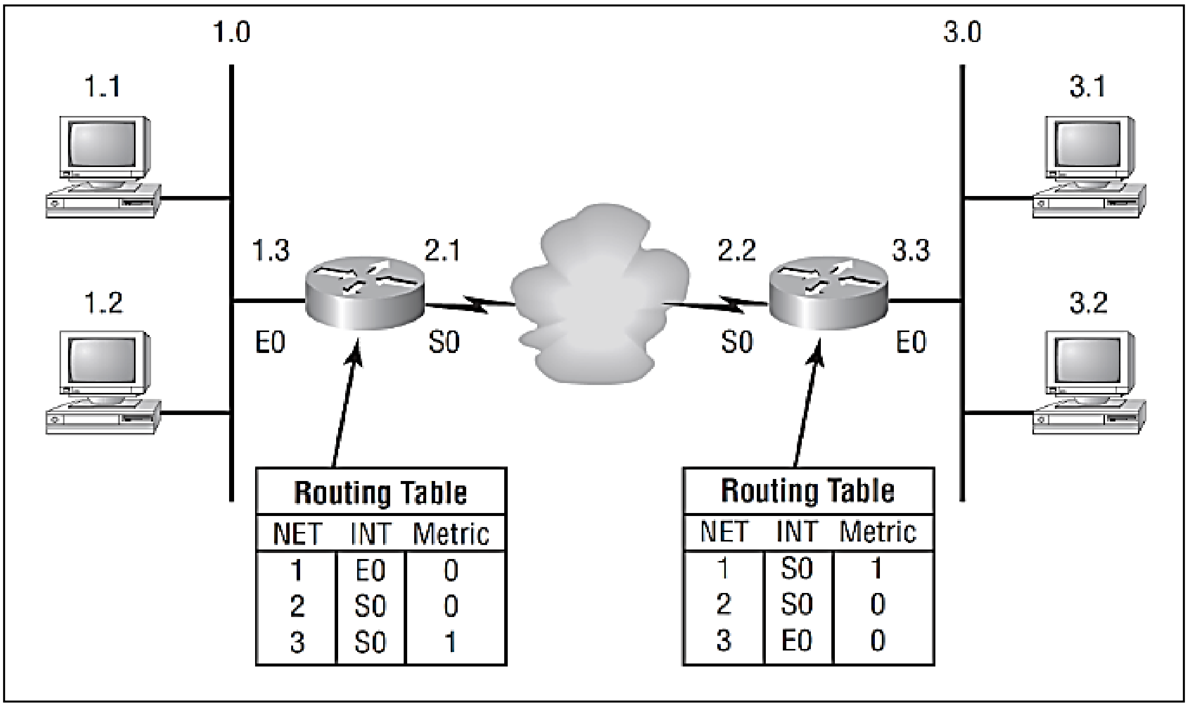

Figure 2.7: Structure Table Of The Router [39].

2.9 ROUTES PROTOCOLS

Routing protocols are used in computer networks to help routers communicate with each other and to determine the best path for forwarding data packets [40]. There are several types of routing protocols, each with their own advantages and disadvantages.

a. Distance Vector Routing Protocol: Distance Vector protocols, such as Routing Information Protocol (RIP), determine the best path to a destination by measuring the distance (hop count) to that destination. They periodically send updates about their routing table to their neighbors. This type of protocol is easy to configure, but can suffer from slow convergence times and routing loops.

b. Link State Routing Protocol: Link State protocols, such as Open Shortest Path First (OSPF), rely on each router having a complete map of the network topology. Each router calculates the shortest path to each destination based on this map. This type of protocol is highly scalable and has fast convergence times, but can be more complex to configure.

c. Hybrid Routing Protocol: Hybrid protocols, such as Enhanced Interior Gateway Routing Protocol (EIGRP), combine the best features of Distance Vector and Link State protocols. They use Distance Vector techniques for intra-area routing and Link State techniques for inter-area routing. This type of protocol is highly scalable, has fast convergence times, and is easy to configure.

d. Border Gateway Protocol: Border Gateway Protocol (BGP) is a complex protocol used for inter-domain routing on the Internet. It allows Internet Service Providers (ISPs) to exchange routing information about their networks and to make decisions about the best path for traffic to flow between them. BGP is a path-vector protocol that takes into account several factors when selecting the best path, including path length, policy, and path attributes.

   Each routing protocol has its own strengths and weaknesses, and the choice of which protocol to use will depend on factors such as the size and complexity of the network, the level of control required, and the types of devices being used. In many cases, a combination of different protocols may be used in different parts of the network to achieve the best balance of performance and efficiency.

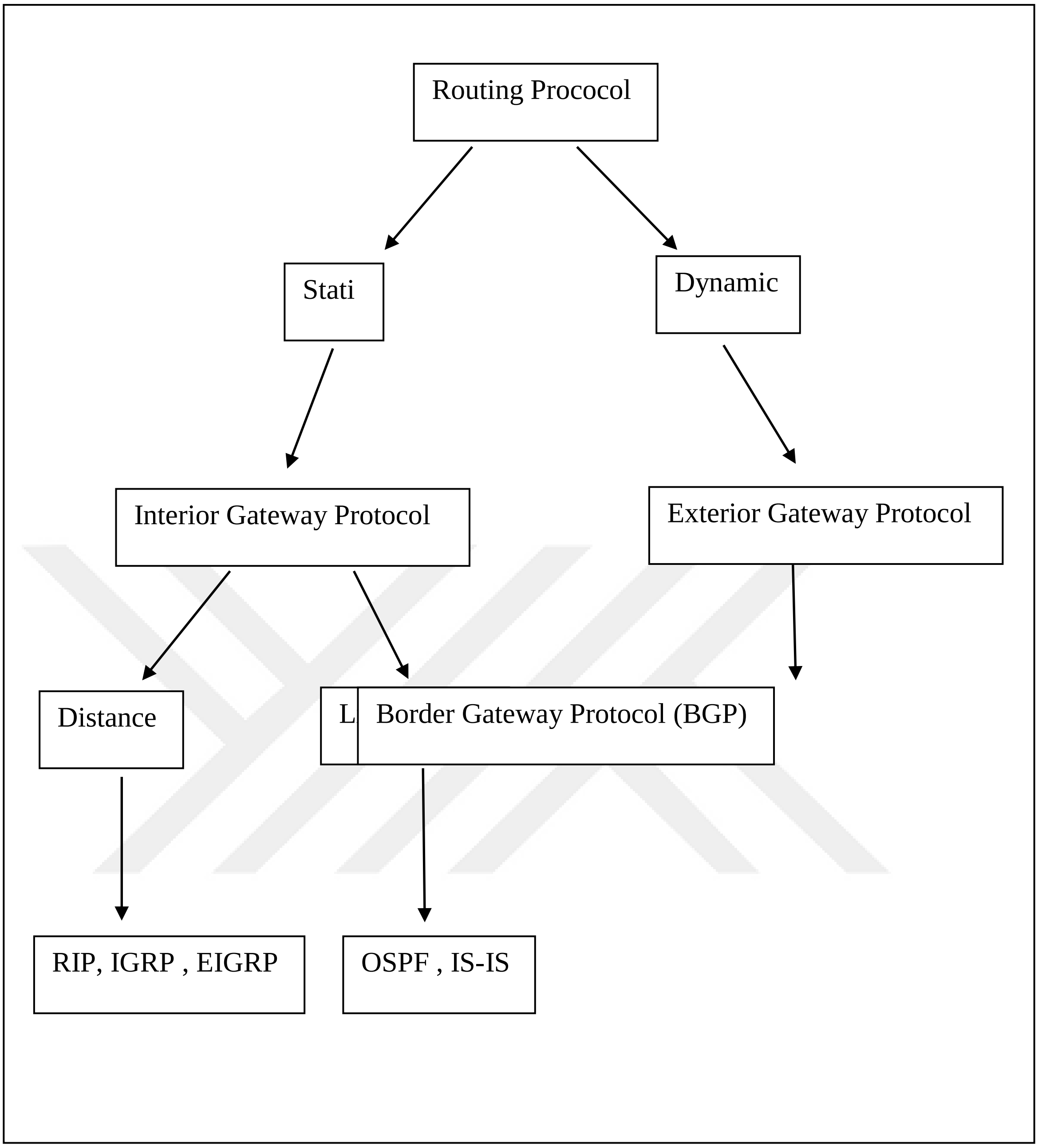

Figure 2.8: Classification Of Routing Protocols [40].

3. METHODOLOGY

3.1 INTRODUCTION

This section explores novel cognitive routing techniques that exhibit significant performance improvements. Some of these include the Shortest Path, the Ant-Colony Route, and the Fastest Path. The study also compares these with existing methods within a multi-level protocol network to demonstrate how they can help improve the transmission of voice and video packets. The study analyzed the effects of different routing algorithms on packet throughput, such as the Ant Colony, Shortest Path, and the Fastest Path, using metrics such as delay count, loss rate, average hop count, and number of packets sent per second. The results of the simulation revealed that implementing recovery methods can improve network performance.

3.2 RESEARCH PROCESS

In today's world, network communication plays a crucial role in the successful operation of businesses and organizations. To facilitate effective and efficient communication, various network protocols have been developed, with MPLS (Multiprotocol Label Switching) being one of the most widely used. This article delves into the details and functions of the MPLS network and examines previous MPLS methods discussed in the literature review. The MPLS network is a high-performance, packet-forwarding technology that provides fast data transfer and traffic management. It operates at the network layer, where it can support different types of traffic, such as voice, video, and data. MPLS works by assigning labels to packets, which are then used to forward them through the network. The labels provide a faster and more efficient way to route packets, and they can be used to implement various network services, such as Quality of Service (QoS) and Virtual Private Networks (VPNs). To better understand the performance of the MPLS network, various parameters must be examined, such as packet loss and transmission rates. A simulation can be used to keep track of these parameters, providing insights into the network's efficiency and effectiveness. In this article, a simulation is used to analyze the performance of the MPLS network, with the lost or transmitted packets being clearly shown. After the creation of the MPLS network, three different routing algorithms are used to route the packets. These algorithms are analyzed, and the resulting network is designed in MATLAB, a widely used simulation tool for network analysis. The results of the simulation are then compared and analyzed to identify any areas of improvement. One key area of improvement is in network recovery. The proposed method for network recovery is introduced, and its effectiveness is compared with the obtained results. The performance improvement of the proposed method is presented graphically, showcasing the significant difference it can make in improving the MPLS network's reliability and stability.

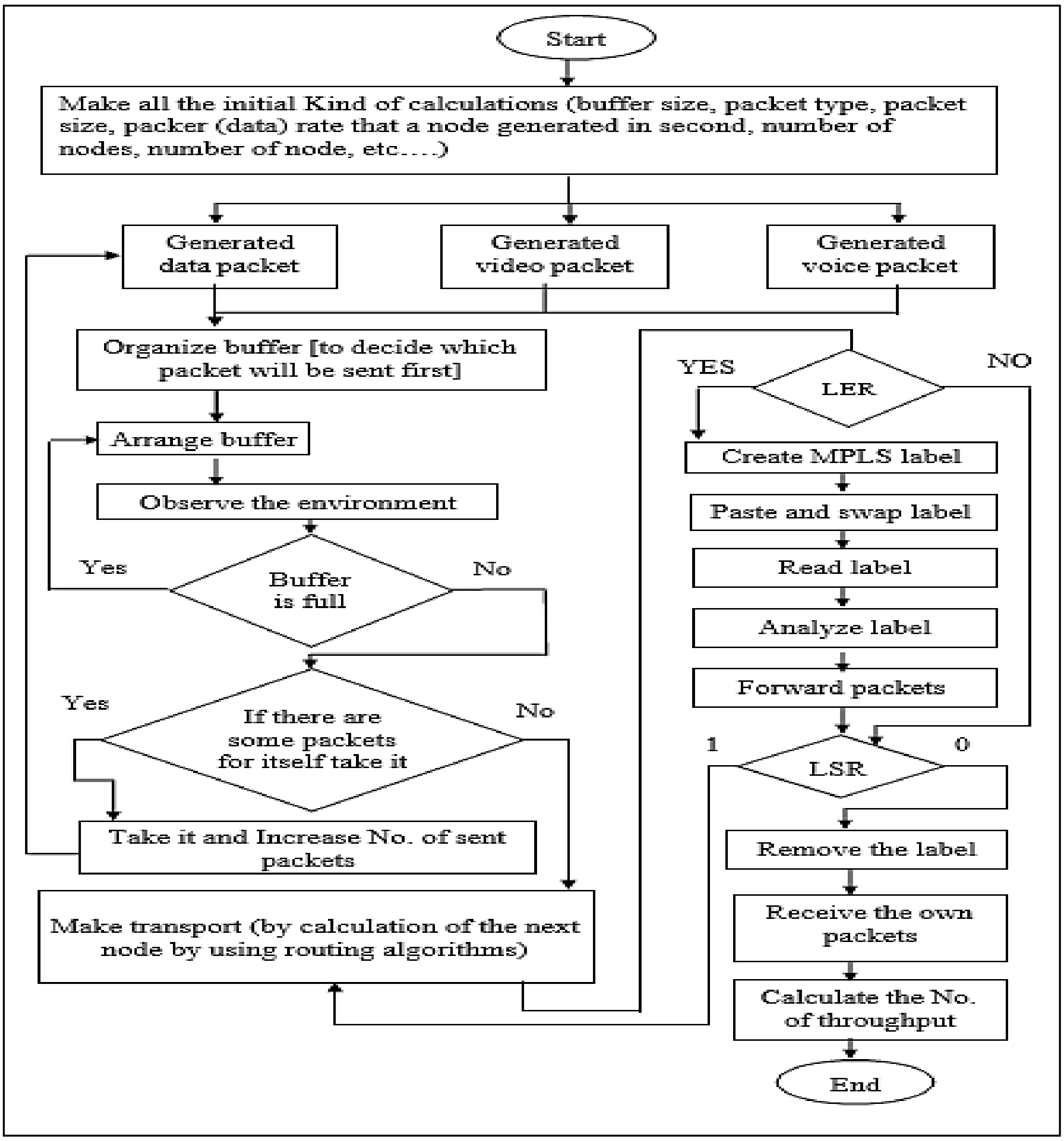

Figure 3.1: Overall The Steps Of Proposed Method.

3.3 PROPOSED ANT COLONY METHOD

An optimization algorithm based on the behavior of ants is called the Ant Colony Algorithm. It is commonly used to solve optimization problems, including finding the optimal path in computer networks. We employed in the computer networks, the Ant Colony Algorithm can be used to optimize the routing of data packets [41]. The goal is to find the most efficient path for each packet through the network, taking into account factors such as bandwidth, latency, and cost. We have applied the Ant Colony Algorithm to optimal path finding in computer networks, a graph representation of the network is first created. Each node in the graph represents a network device, and the edges represent the connections between the devices. The pheromone levels on each edge are then used to guide the ants as they search for the optimal path. The Ant Colony Algorithm works by simulating the behavior of real ants. The ants move through the graph by following a set of simple rules. Each ant deposits pheromone on the edges it traverses, and the amount of pheromone deposited is proportional to the quality of the solution. As the ants move through the graph, they communicate with each other using pheromone trails. Ants are more likely to follow paths with higher pheromone levels, so over time, the pheromone trails converge on the optimal solution, leading to more efficient routing. The pheromone trails evaporate over time, allowing the algorithm to explore new paths. This helps prevent the ants from getting stuck in local minima and encourages the algorithm to continue exploring new solutions. The Ant Colony Algorithm is particularly effective in situations where traditional optimization methods may not work well, such as when the problem space is very large or complex. By mimicking the behavior of real ants, the algorithm is able to efficiently explore large and complex solution spaces, leading to better performance and cost savings. Furthermore, the Ant Colony Algorithm is a powerful optimization technique that can be used to find the optimal path in computer networks. By following the behavior of real ants, the algorithm is able to efficiently explore the solution space and converge on the best solution. The Ant Colony Algorithm is a metaheuristic optimization algorithm that mimics the behavior of ants in nature to find the optimal solution to a given problem. In the case of MPLS routing, the problem is to find the most efficient path for data packets through the network. The relationship between MPLS and the Ant Colony Algorithm is shown in Figure 3.2, which illustrates how the algorithm is applied to find the optimal path.

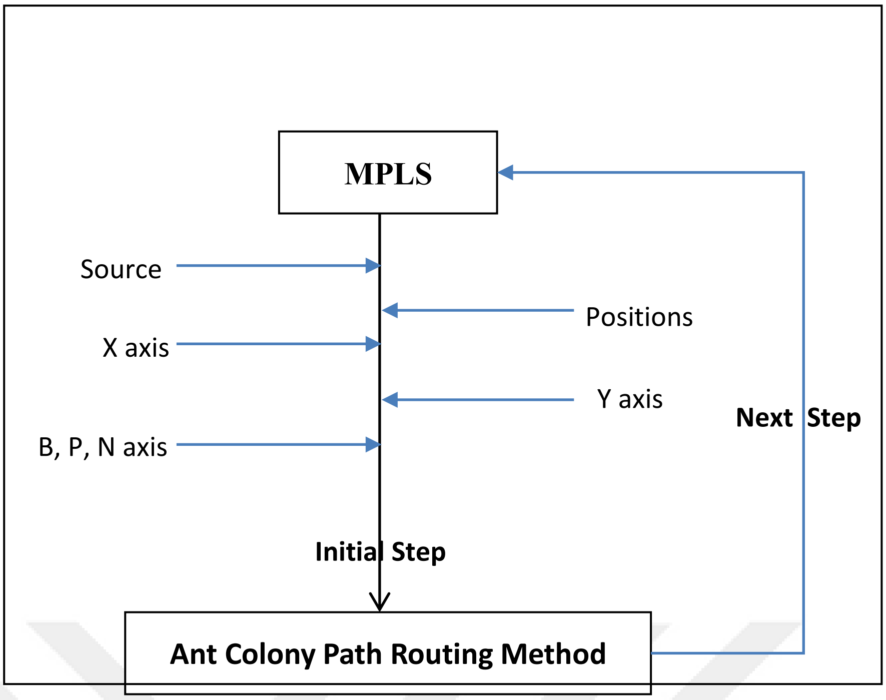

Figure 3.2: The Combination Of The MPLS And Anti Colony Method.

3.4 OPTIMAL FASTEST PATH METHOD

To determine the fastest path in a network, the first step in the optimization process is to create a mathematical model that can identify the ideal locations of relay nodes, and then use that model to perform the optimization. Next, a routing algorithm is designed to discover the optimal path. Through research, it has been shown that this new routing algorithm is capable of successfully transporting packets in a very short amount of time, particularly in situations where there is a high density of nodes. The Fastest Path Algorithm is a network routing algorithm that can be applied to optimize the routing of data packets through an MPLS network. This algorithm is designed to find the fastest path between two points in the network, taking into account factors such as available bandwidth and latency [42]. To apply the Fastest Path Algorithm to MPLS routing, a graph representation of the network is created, with nodes representing network devices and edges representing the connections between them. Each edge is assigned a weight that represents the time it takes to traverse the edge, taking into account the characteristics of the MPLS network. The specific math used in the Fastest Path Algorithm depends on the implementation of MPLS used in the network. For example, if the network uses CR-LDP, the weight of an MPLS label-switched path can be calculated using the formula:

Weight = (1/BW) + Delay

Where BW is the available bandwidth and Delay is the delay of the LSP.

Once the weights of the edges in the graph have been calculated, the algorithm uses Dijkstra's algorithm to find the shortest path between the source and destination nodes. This path takes into account the specific characteristics of the MPLS network, such as available bandwidth and latency, and is optimized to provide the fastest possible path for data packets. The Fastest Path Algorithm is an effective method for optimizing network routing, including in MPLS networks. By taking into account the specific characteristics of MPLS, the algorithm can find the most efficient path for data packets, leading to better network performance and cost savings. This makes it a valuable tool for researchers and network engineers looking to optimize the performance of their MPLS networks. Within this framework, we presume that each node possesses the ability to obtain and utilize the subsequent pieces of information: its personal location, the location of the packet's destination, the next step involves implementing the fastest algorithm for packet transmission on each relay node in the system. This algorithm will then determine the path to the next hop forwarding node. The transmission radius is determined by

$$\boldsymbol{x} * \approx \left(\frac{\boldsymbol{p}}{\boldsymbol{N}(\boldsymbol{e}^{\boldsymbol{a}} - \mathbf{1})}\right)^{\frac{\mathbf{1}}{\boldsymbol{a}}} \tag{3.1}$$

taking into account the following equation:

The packet is sent to ud if x is greater or equal to its value. The ui algorithm then takes into account the link's C capacity to calculate the transmit and receive rate [23]. Doing so will prevent reception issues while ensuring that the packet is received at the right time. In this case, one additional relay node might be required if x is less than one uiud. The algorithm starts by determining the ideal next-hop node's location, takes into account the linked capacity, and sends the packet to the appropriate destination

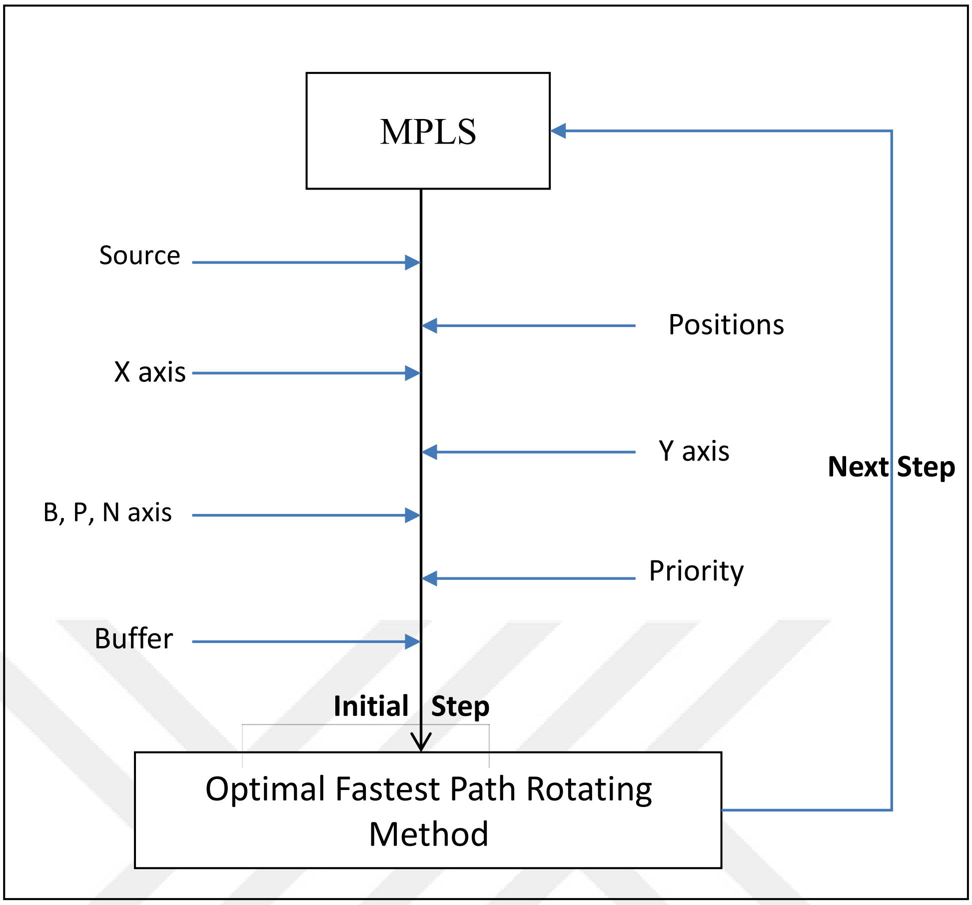

Figure 3.3: The Diagram Illustrates The Combination Between MPLS And The Fastest Path Method.

3.5 OPTIMAL SHORTEST PATH METHOD

The most frequently used metrics in IP routing are destination address and delay or hop-count. A variety of algorithms are also good for determining the shortest paths to a destination. However, the Floyd-Warley algorithm, which is suitable for regular graphs, is not suitable for WLANs due to its lack of handshaking [24]. The performance of short-path routing algorithms is often evaluated to see how they can reduce energy consumption. A cost metric can also be calculated by taking into account the provided equation.

$$a_{ij}^{k} = \frac{1}{cij} \tag{3.2}$$

The total capacity of the link, denoted by cij, is used to calculate the cost metric in shortest path routing algorithms. An alternative to shortest path routing is minimum hops routing, which selects the path with the minimum number of hops as the metric [43]. The lengths of the shortest route to each destination are maintained by the nodes, and updates are only sent to their neighbors. The algorithm known as the Dinkstra Algorithm is used to route connections between the sources and the other nodes in a network. It is commonly used in networks like OSPF, which is designed to monitor the status of the network's various nodes. The diagram of the algorithm in Figure 3.4 shows the function's structure.

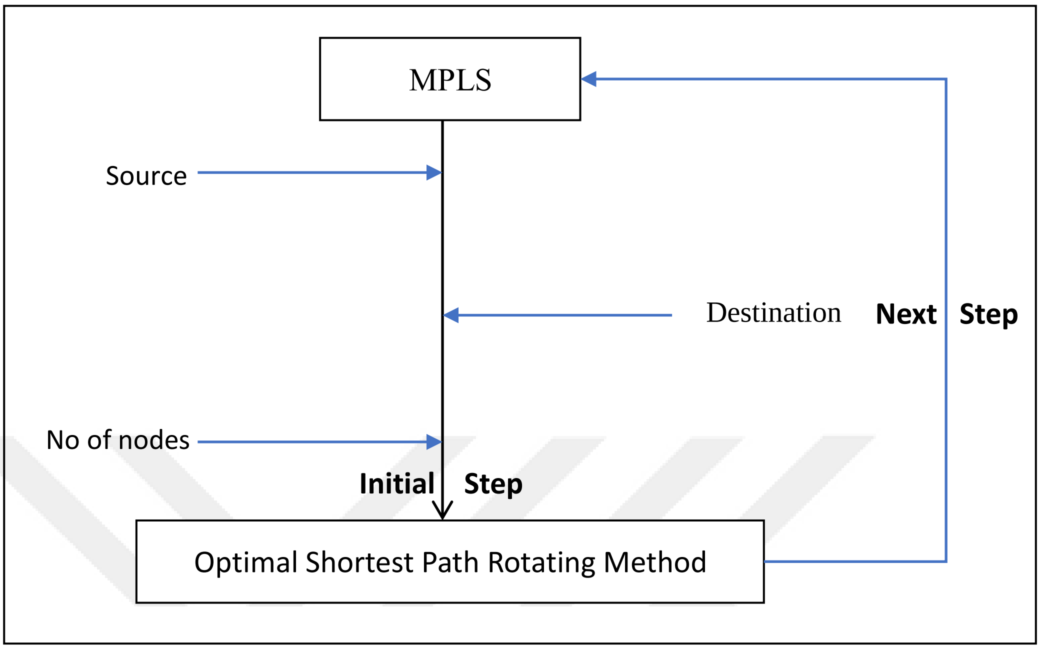

Figure 3.4: The Diagram Illustrates The Combination Between MPLS And The Shortest Path Method.

3.6 RESEARCH MAPPING FOR OPTIMAL ROUTING ALGORITHM

The different methods used in this study, three proposed methods have been analyzed and compared to their outputs. The method of determining the optimal routing algorithm for the supply of a network has also been performed simultaneously with and in the same circumstances. The impact of different routing methods on the system's outcomes results are analyzed as shown in the figure 3.5. The findings of this study can be shown in graphical representations, which highlight the routing methods used in the simulation.

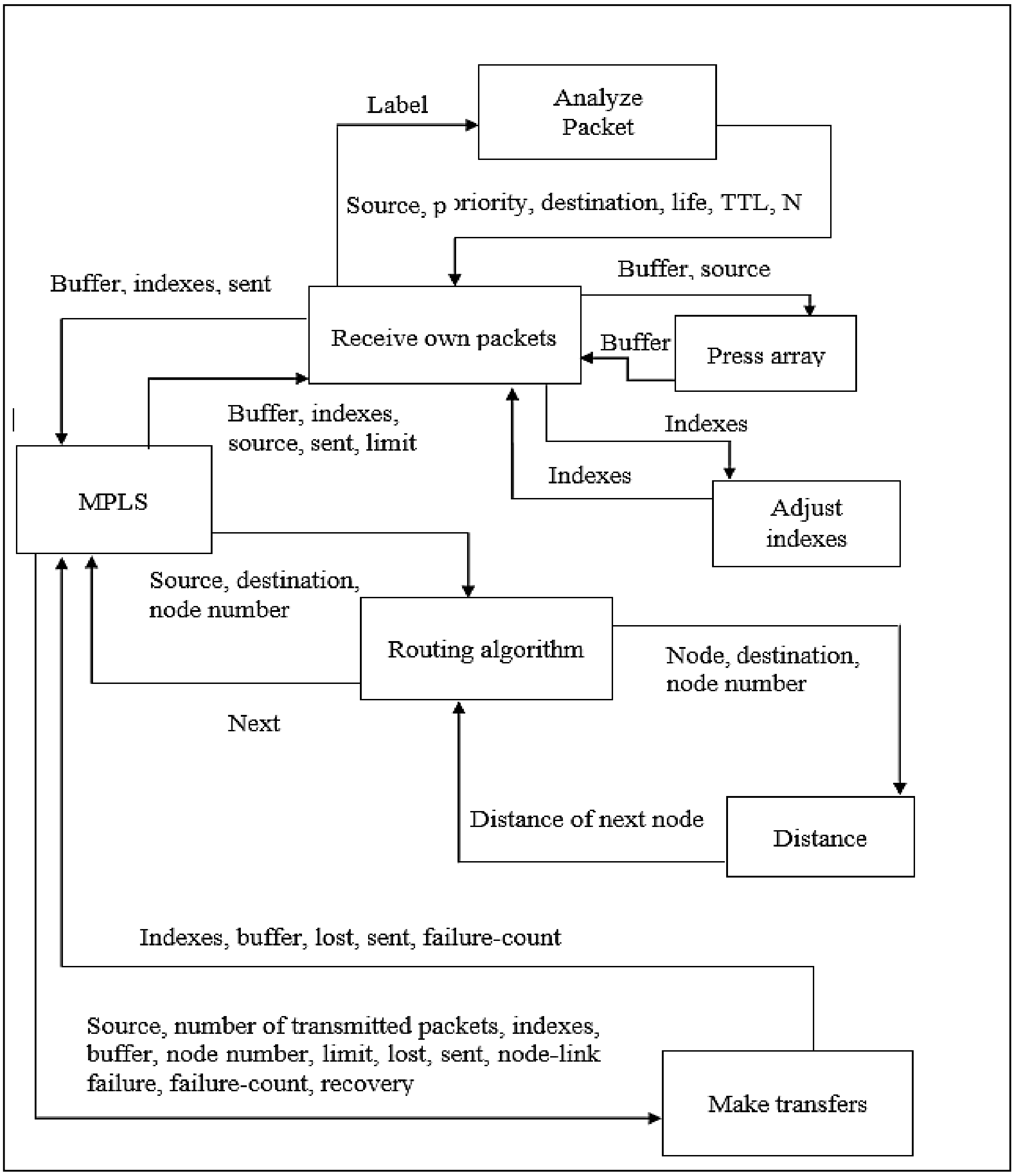

Figure 3.5: The Process Of Three Routing Algorithms.

3.7 DIJKSTRA'S ALGORITHM

Dijkstra's algorithm is a graph search algorithm that solves the single-source shortest path problem for a weighted graph. It finds the shortest path from a source vertex to all other vertices in the graph. The algorithm works by assigning a tentative distance value to every vertex, which is the smallest known distance from the source vertex to that vertex. As the algorithm progresses, the tentative distance values are updated and improved until the shortest path to every vertex is found.

The algorithm uses a priority queue to select the vertex with the smallest tentative distance value as the current vertex. It then examines each of the neighbors of the current vertex and updates their tentative distance values if a shorter path to them is found. This process continues until the algorithm has found the shortest path to every vertex in the graph.

In mathematical notation, the algorithm can be described as follows:

a. Set the distance value of the starting vertex to 0, and all other vertices to infinity.

b. Add all vertices to a priority queue, sorted by their distance value.

c. While the priority queue is not empty: a. Extract the vertex with the smallest distance value from the priority queue. b. For each neighbor of the current vertex: i. Calculate the tentative distance to the neighbor through the current vertex. ii. If the tentative distance is less than the neighbor's current distance value, update the neighbor's distance value in the priority queue.

d. When all vertices have been visited, the shortest path to each vertex has been found.

In the context of route path in PMLS, Dijkstra's algorithm can be used to find the shortest path between two points on a map. The vertices in the graph represent locations on the map, and the edges between them represent the distances or travel times between those locations. By applying Dijkstra's algorithm to this graph, we can find the shortest path between two locations and provide directions for the route that follows that path.

The mathematical notation for Dijkstra's algorithm is as follows:

Let $G = (V, E)$ be a weighted graph with vertex set V and edge set E, where each edge (u,v) has a weight $w(u, v)$ representing the cost of traveling from vertex u to vertex v.

Let s be the starting vertex for which we want to find the shortest path to all other vertices.

Let $dist[v]$ be the tentative distance from the starting vertex s to vertex v.

Let S be a set of visited vertices, initially empty.

Let Q be a priority queue of vertices, sorted by their tentative distances.

Then the algorithm proceeds as follows:

a. Initialize $dist[s] = 0$ and $dist[v] = \text{infinity}$ for all other vertices v.

b. Add all vertices to Q.

c. While Q is not empty: a. Remove the vertex u with the smallest dist[u] from Q. b. Add u to S. c. For each neighbor v of u: i. Calculate the tentative distance to v through u as alt = dist[u] + w(u, v). ii. If alt < dist[v], update dist[v] = alt. d. Reorder Q based on the updated dist values.

d. When all vertices have been visited, the shortest path to each vertex has been found. Note that this algorithm can be adapted to handle directed and undirected graphs, as well as graphs with negative edge weights (although additional precautions must be taken in the latter case to avoid infinite loops).

Table 3.1: Represent Of Dijkstra's Algorithm Structure

Path	**Short path**	**BASE**
A —> E —> B	4	A —> B
A —> E —> B —> C	6	A —> C
A —> E —> D	5	A —> D
A —> E	3	A —> E

As illustrates in table 3.1, and figure 3.6, that in a weighted digraph, given a set of vertices with edge weights that are non-negative, the shortest path weights for each of the vertices are found from the source.

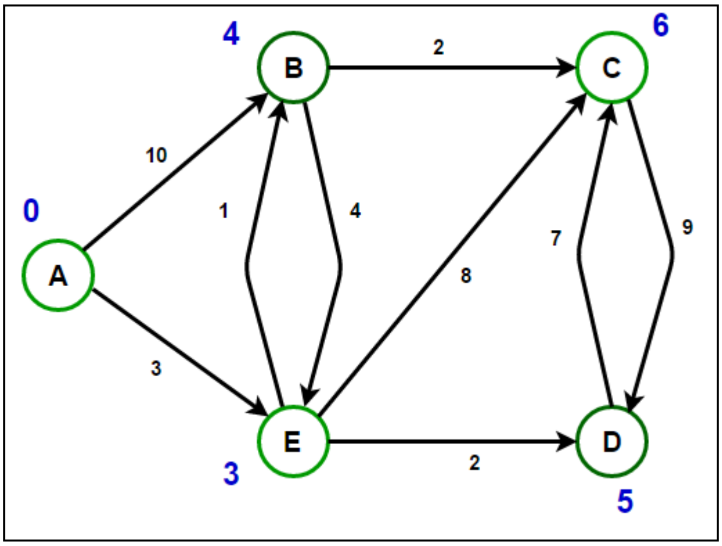

Figure 3.6: Represent Of Dijkstra's For Path Route [43].

The source of the digraph is vertex A, and we'll start relaxing its neighbours. Update the distance of the vertices B and E to 10 and 3 respectively as shown in the figure 3.7.

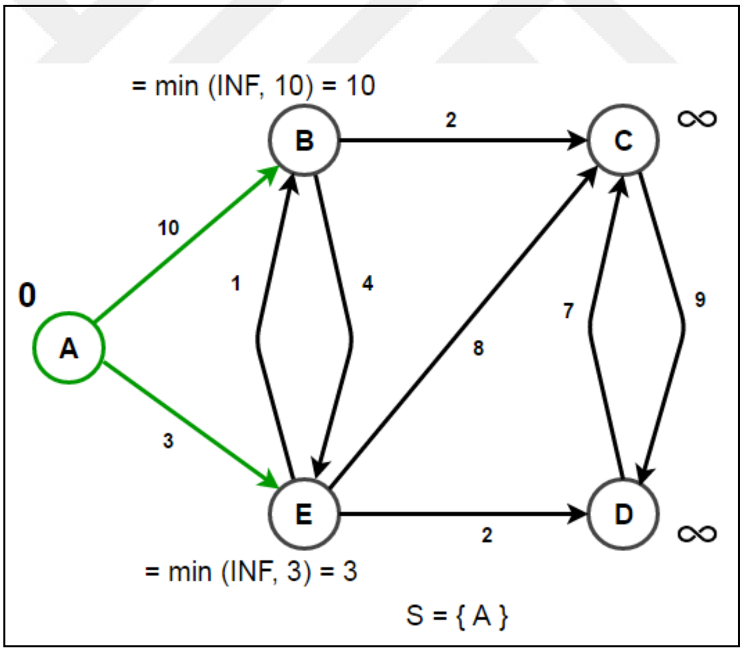

Figure 3.7: The Source Of The Digraph Is Vertex A [46].

3.8 ROUTE INDTIFGATION AND DISCOVERY

To begin, the process of relaxing the neighbors of source node A should be initiated. This involves adjusting the distances of neighboring nodes to enable them to communicate with other networks. For example, by increasing the distance of source node B from 10 to 20, it can connect with its neighbor node E. Similarly, by increasing the distance of source node C from 2 to 3, it can reach its neighbor node F. When a node needs to communicate with other networks, it first checks the routing table to find an available path. This route may be newly created or updated if necessary. If the routing table does not contain a path to the source IP address, a reverse route is established.

It is important to note that the Inverted Metrical Value (IMV) in the routing table is only updated when the sequence number of the source is higher than that of the destination. The IMV is a unique feature that differs from the previous Integrated Metric Value (IMV) and is calculated using the equation 3.3 in this study.

$$\boldsymbol{M_N} = \left((\boldsymbol{C}/\boldsymbol{C_{PE}}) + (\boldsymbol{T}/\boldsymbol{T_P}) + (\boldsymbol{H}/\boldsymbol{H_P})\right) \tag{3.3}$$

The equation 3.3 is utilized to calculate the previous-integrated-metric-value. By plugging in the appropriate variables and solving for the equation, the previous-integrated-metric-value can be determined.

$$\boldsymbol{M_P} = \boldsymbol{C_{PE}} + \boldsymbol{T_P} + \boldsymbol{H_P} \tag{3.4}$$

To initiate the route discovery process, the first step is to check the routing table for information about the destination. If the reference number in the RREP (Route Reply) message is greater than the number of copied copies of the destination sequence, a forward route is established. However, if the sequence numbers are the same and the route has already been active, it is considered invalid and no longer exists.

Figure 3.8 provides a clear illustration of the route discovery process in the form of a flow chart. By following the steps outlined in the flow chart, the process of discovering and establishing a valid route becomes more streamlined and efficient.

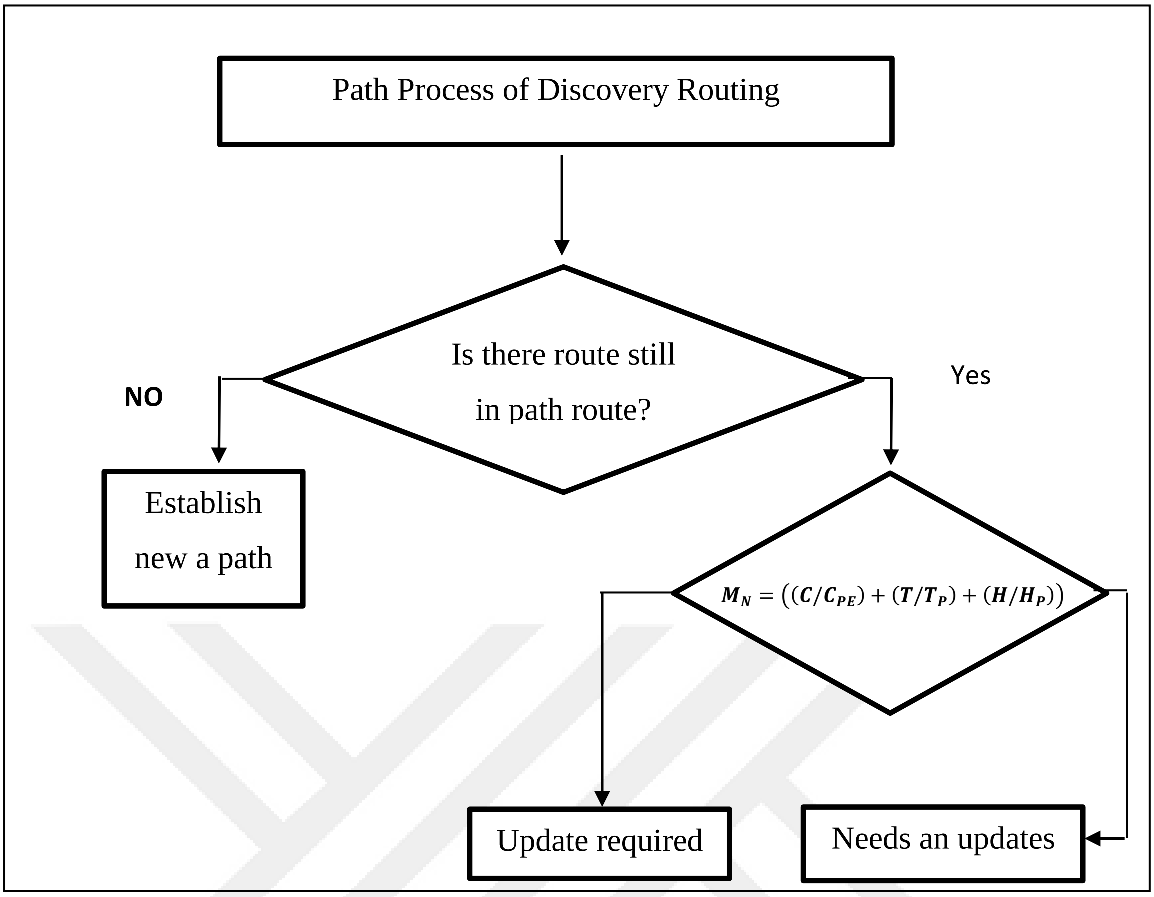

Figure 3.8: Flow Discovery Of Routing Path.

3.9 CHAPTER SUMMARY

Chapter 3 of the methodology discusses new cognitive routing techniques that can enhance the performance of multi-level protocol networks in transmitting voice and video packets. The section compares these novel methods, such as the Shortest Path, Ant-Colony Route, and Fastest Path, with existing ones. The chapter evaluates the impact of different routing algorithms on packet throughput and examines metrics such as delay count, loss rate, average hop count, and the number of packets sent per second. The results of the simulation indicate that integrating recovery methods can enhance the network's performance.

4. PERFORMANCE EVALUATION

4.1 INTRODUCTION

The three algorithms suggested in this study are Ant colony, the shortest path, and the fastest path. They are designed to provide a fast and simple way to route packets. The goal of this simulation is to analyze the various performance parameters of the algorithms that are utilized in the network. The continuous transmission of packets using each of the different routing algorithms has been performed for a total of 100000 unit of time for both long and short-term implementations. The program's results, as well as the analysis of the algorithms' performance, are then evaluated and compared.

4.2 RESULTS

The three algorithms suggested in this study are Ant colony, the shortest path, and the fastest path. They are designed to provide a fast and simple way to route packets. The goal of this simulation is to analyze the various performance parameters of the algorithms that are utilized in the network. The continuous transmission of packets using each of the different routing algorithms has been performed for a total of 100000 unit of time for both long and short-term implementations. The program's results, as well as the analysis of the algorithms' performance, are then evaluated and compared.

4.2.1 Optimal Fastest Path Method

Regarding Optimal Fastest Path Method, the Each node generates 6,00012 packets and has a buffer capacity of 10. Due to implementation errors, there are 895 errors in the link or node. Despite this, the algorithm was able to successfully transmit nearly 59,000 packets. Two implementations of the algorithm are shown in Figures 4.1, 4.2,4.3,4.4,4.5, and 4., long implementation and a short implementation. The throughput of the long implementation is 3.77369e, while the short implementation's throughput is 3.66454e07. In other words, the long implementation has a slightly higher throughput than the short implementation, but both have achieved a significant amount of successful packet transmission despite the implementation errors.

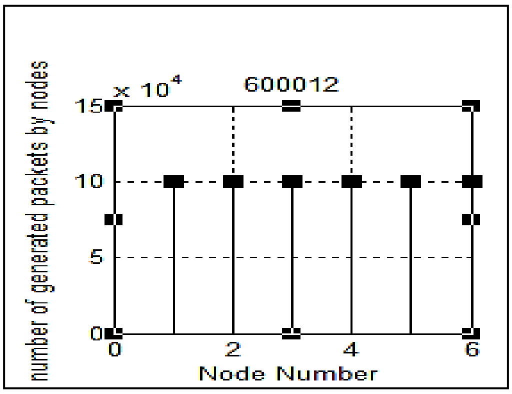

Figure 4.1: Packets For Each Node.

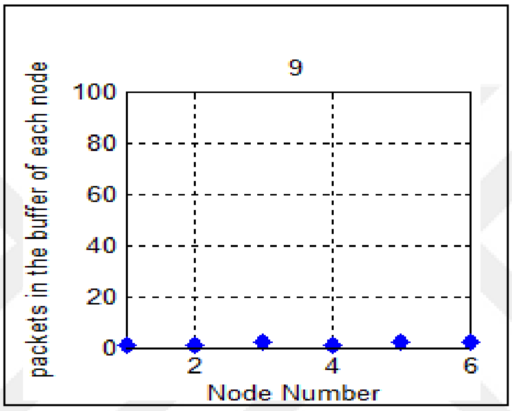

Figure 4.2: Node And Buffer Companion Relation.

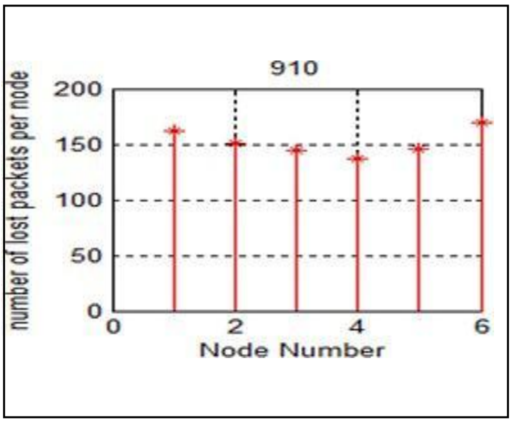

Figure 4.3: Missed Packets In Each Node.

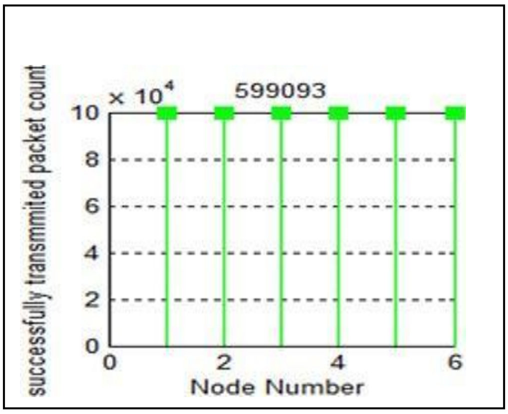

Figure 4.4: Succeed Packet Ration In Each Not Transmitted.

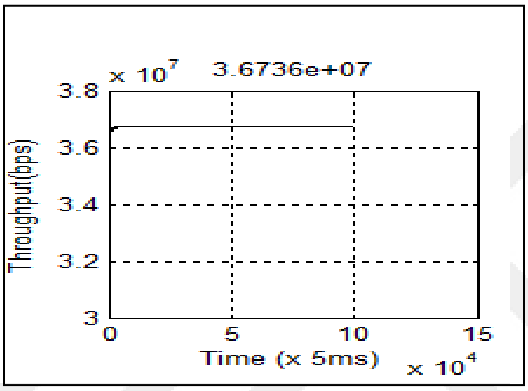

Figure 4.5: Outcomes Of Long Simulation.

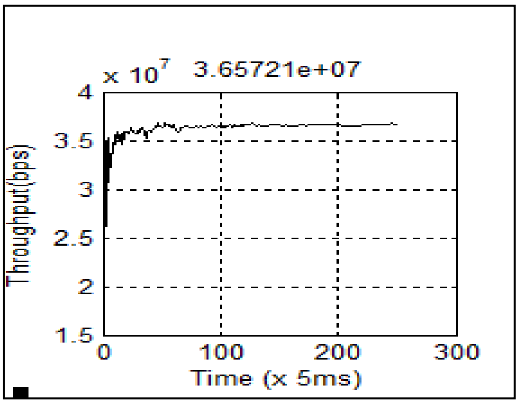

Figure 4.6: Outcomes Of Short Simulation.

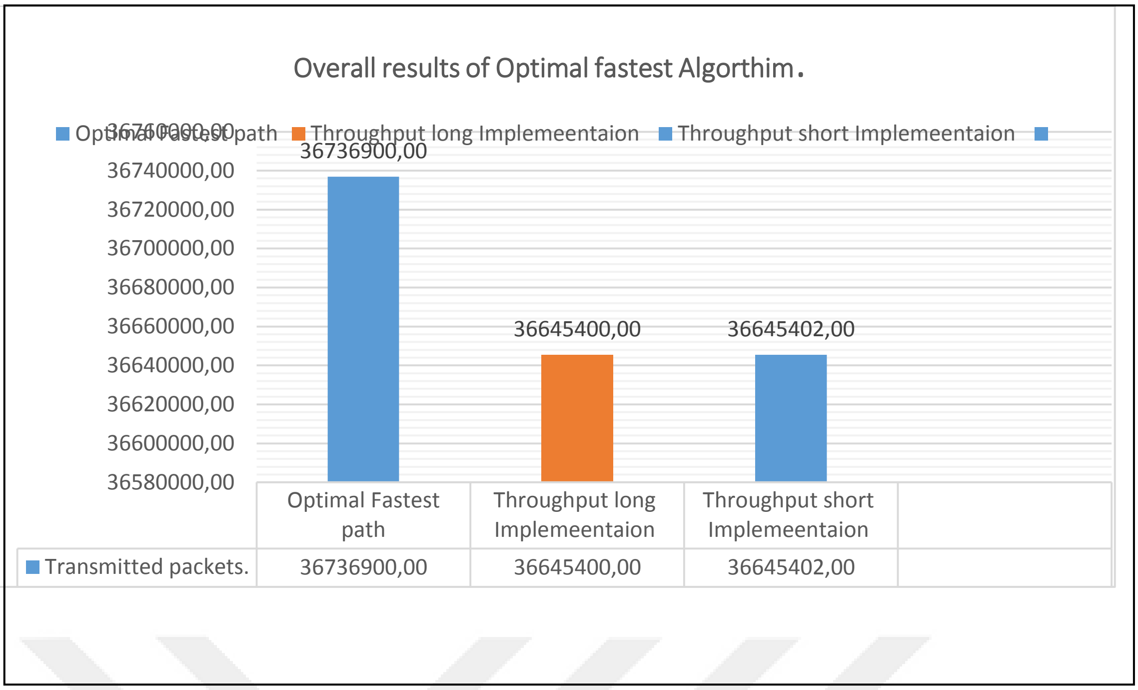

Figure 4.7: Overall Results Of Optimal Fastest Algorithm.

4.2.2 Optimal Shortest Path Algorithm

Figures 4.8, 4.9, 4. 10, and 4.11 is likely a visualization or graph that displays various metrics related to a network communication scenario. The graph appears to show the packet count and buffer size for each node involved in the communication, as well as information about link and node errors.

The "optimal shortest path" likely refers to the most efficient route for data to travel through the network to reach its destination. By analyzing this information, network administrators can identify potential bottlenecks or areas of the network that may need improvement. In addition to this, the graph also displays information about the long-term and short-term throughputs of the algorithm. Throughput refers to the amount of data that can be transmitted over a network within a certain amount of time. The long-term implementation of the algorithm shows a throughput of 3.6736e+07. This means that over an extended period of time, the network is capable of transmitting roughly 36.7 million data packets per second. The short-term implementation, on the other hand, has a slightly lower throughput of 3.61e+07. This suggests that while the network is still highly efficient, there may be some variability in the amount of data that can be transmitted in shorter periods of time.

Overall, Figures likely provides valuable insight into the performance and efficiency of a network, allowing administrators to identify areas for improvement and ensure that data can be transmitted quickly and reliably.

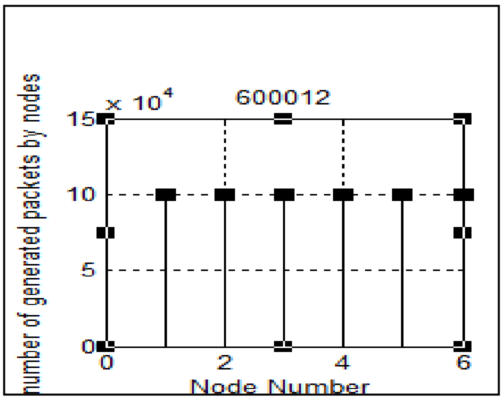

Figure 4.8: Node And Buffer Companion Relation

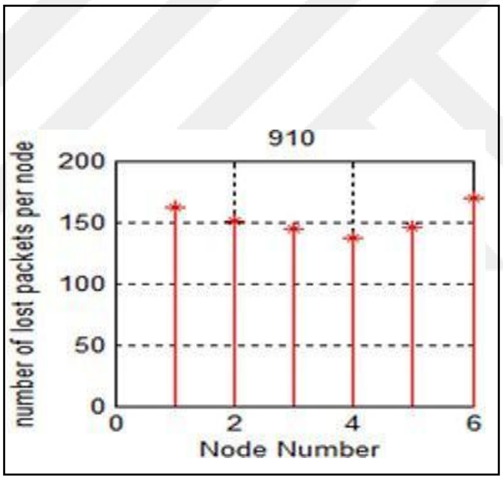

Figure 4.9: Packets For Each Node.

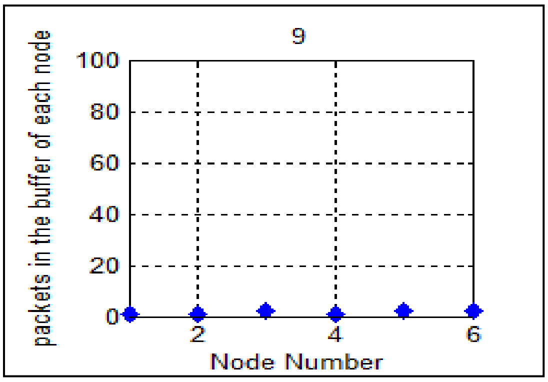

Figure 4.10: Missed Packets In Each Node.

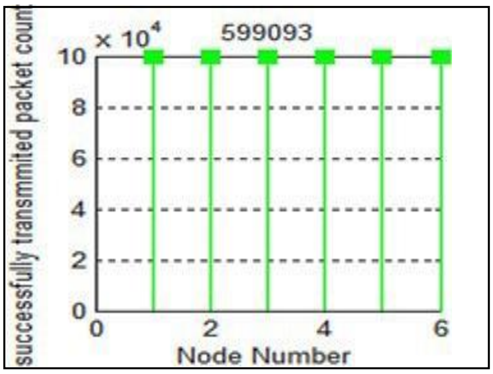

Figure 4.11: Succeed Packet Ration In Each Not Transmitted.

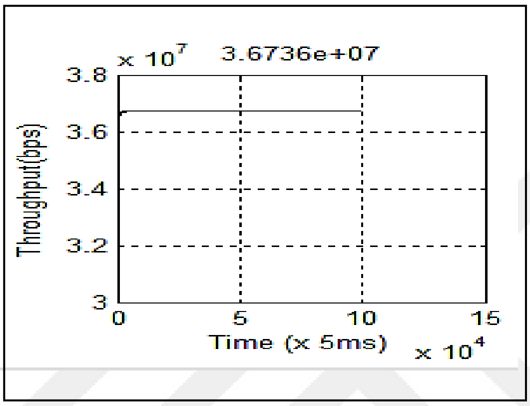

Figure 4.12: Outcomes Of Long Simulation.

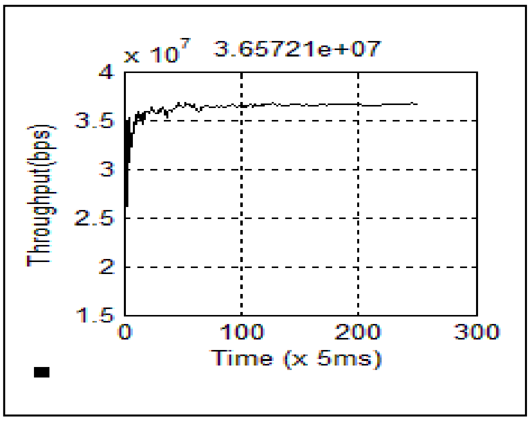

Figure 4.13: Outcomes Of Short Simulation.

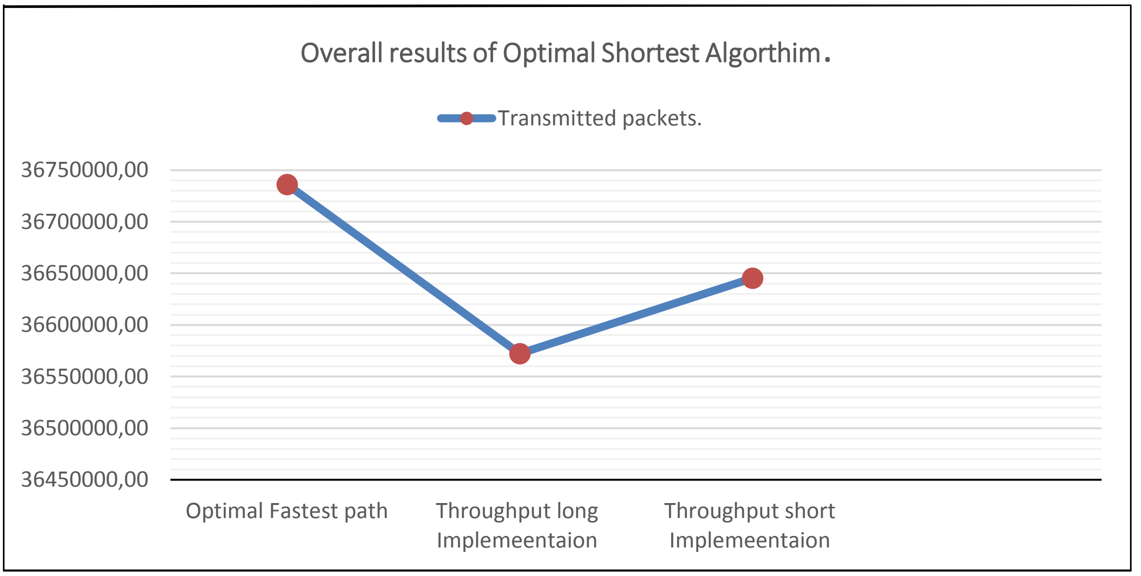

Figure 4.14: Overall Results Of Optimal Shortest Algorithm.

4.2.3 Proposed Ant Colony Method

The text is describing the performance of an ant colony algorithm for data transmission. The algorithm was run on a network of nodes and generated a total of 485,338 individual packets per node, with a total buffer size of 561. Out of these packets, 483,592 were successfully transmitted. This means that almost 48 billion packets were received and sent by the algorithm. In addition to the number of packets, the text provides two measures of the algorithm's throughput. Throughput refers to the rate at which data can be transmitted through a network. The long-term implementation of the algorithm has a throughput of 2.796536e+07, while the short-term implementation has a throughput of 2.93408e. These values are expressed in scientific notation, where "e+07" means "multiplied by 10 to the power of 7" and "e" means "multiplied by 10 to the power of 0. The figures 4.15, 4.16,4.17,4.18, 4.19, and 4.20, that is mentioned likely shows a graphical representation of the algorithm's throughput over time. The text suggests that the short-term implementation of the algorithm has a slightly higher throughput than the long-term implementation.

Overall, the text provides a quantitative assessment of the performance of an ant colony algorithm for data transmission. It reports on the number of packets generated, transmitted, and successfully received by the algorithm, as well as measures of its throughput in both the short and long term.

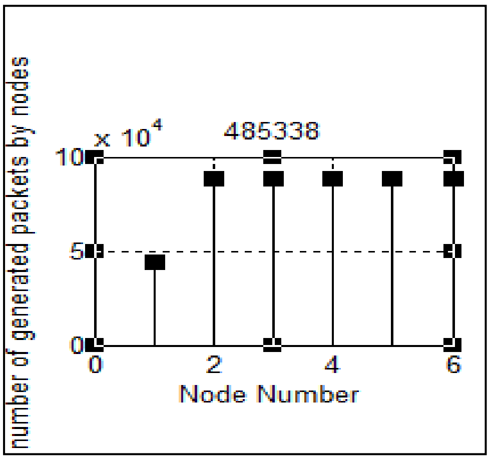

Figure 4.15: Packets For Each Node.

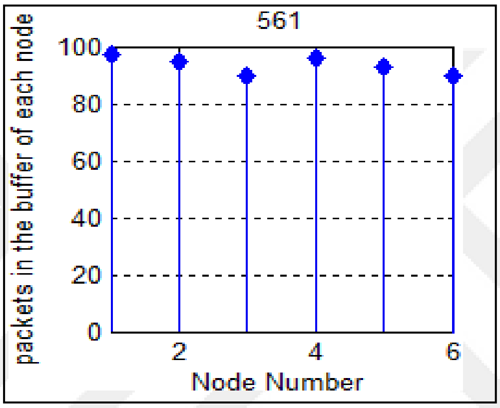

Figure 4.16: Node And Buffer Companion Relation.

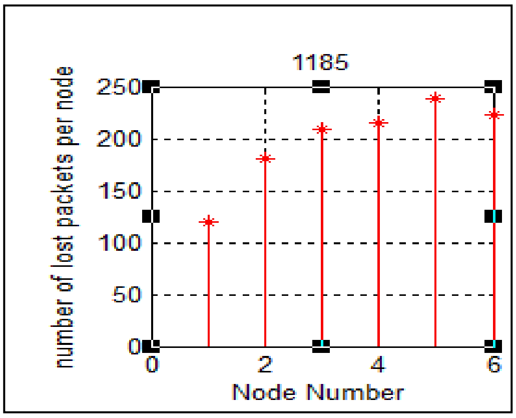

Figure 4.17: Missed Packets In Each Node.

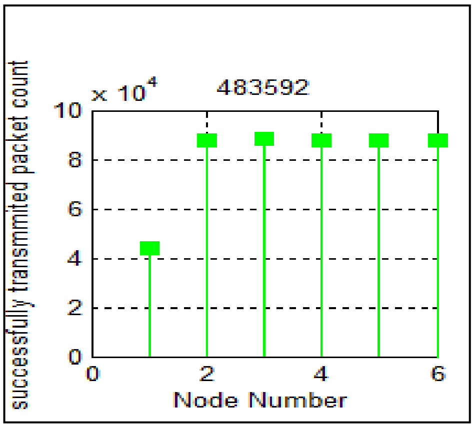

Figure 4.18: Succeed Packet Ration In Each Not Transmitted.

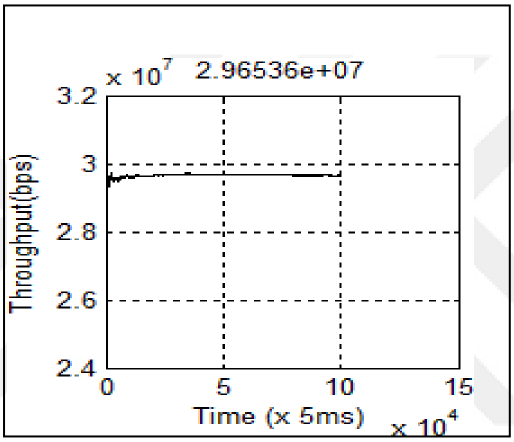

Figure 4.19: Outcomes Of Long Simulation.

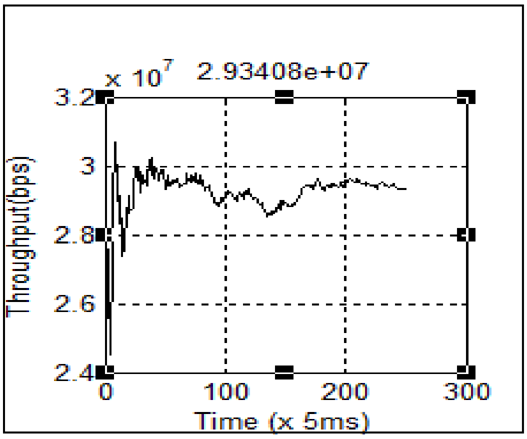

Figure 4.20: Outcomes Of Short Simulation.

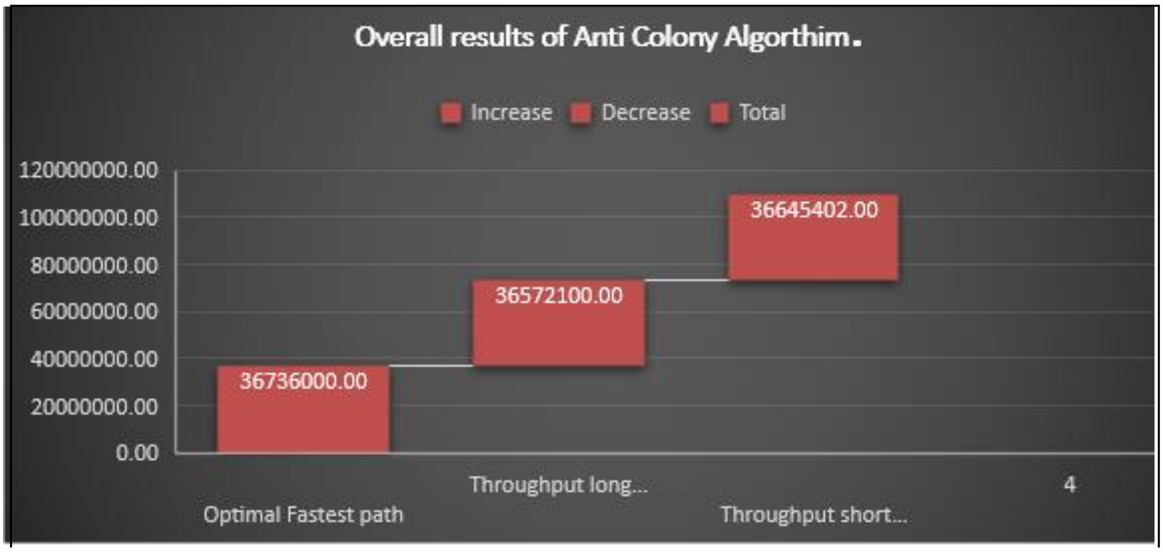

Figure 4.21: Overall Results Of Anti-Colony Algorithm.

4.3 EVALUATION AND COMPARISON OF THREE PROPOSED METHODS

The performance of three algorithms is evaluated using two different run modes: long time and short time. These modes are utilized to analyze the throughput of the algorithms. The method that is ant colony exhibits a higher number of lost packets compared to the other two. Additionally, the shortest and fastest algorithm sends a greater number of packets than the other two algorithms. As a result, the ant colony algorithm has achieved best throughput when compared to the other two. Conversely, the shortest path algorithm performs similarly to the fastest algorithm, both of which outperform the ant colony algorithm. Figure 4.22, provides a graphical representation of the distinct differences between the three methods. And the Figure 4.23 and 4.24 display performance metrics of different algorithms The "fastest algorithm" is likely an algorithm that can perform a given task in the shortest amount of time, while the "shortest path algorithm" is an algorithm that finds the shortest path between two points in a network or graph. . The Ant colony algorithm is a third algorithm being compared in this study. The statement "the fastest algorithm is more efficient than the shortest path one" suggests that the algorithm that can perform the task in the shortest amount of time is more efficient than the one that finds the shortest path. This may be because the fastest algorithm can complete more tasks in the same amount of time as the shortest path algorithm, thus increasing efficiency.

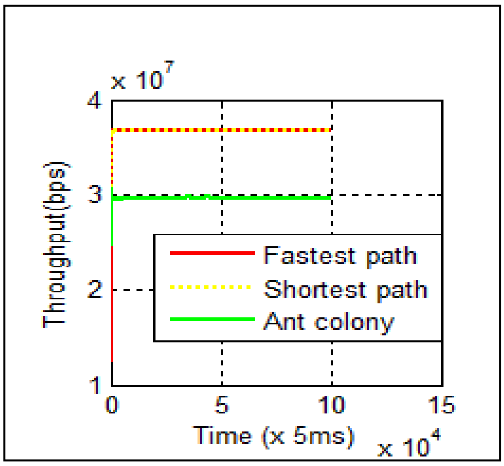

Figure 4.22: Graphical Representation Of The Distinct Differences Between The Three Methods In Terms Of Throughput Of Long Simulation.

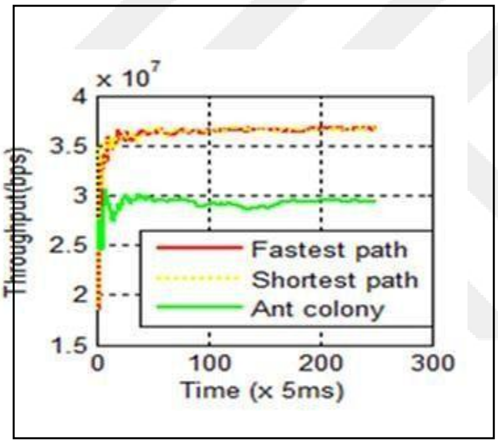

Figure 4.23: Graphical Representation Of The Distinct Differences Between The Three Methods In Terms Of Throughput Of Short Simulation.

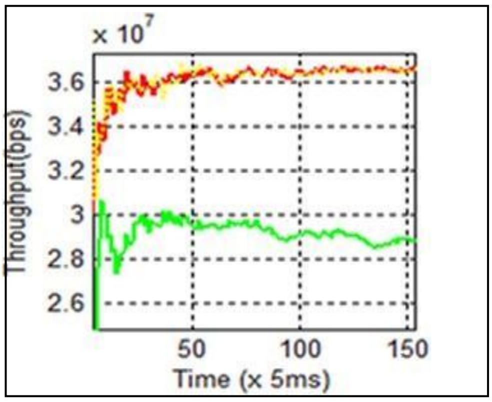

Figure 4.24: Graphical Representation Of The Distinct Differences Between The Three Methods In Terms Of Throughput Of Short Simulation Concentrate in.

The study also found that the Ant colony algorithm was the weakest of the three algorithms. This may mean that the Ant colony algorithm had a lower performance metric than the other two algorithms, such as slower completion time or higher error rate. The study also found that the throughput, or the amount of data transmitted over a network, of the fastest and shortest path algorithms was higher than that of the other two algorithms. One factor that may have influenced the performance of the fastest algorithm was the low value of lost packets, meaning that the algorithm was able to transmit data with high accuracy. Similarly, the high number of transmitted packets by the shortest path algorithm may have contributed to its higher throughput As shown in the figures below 4.25, 4.426,

Furthermore, these findings suggest that the choice of algorithm can have a significant impact on performance metrics such as efficiency and throughput. It is important to consider the specific requirements and constraints of a task when selecting an algorithm, as different algorithms may perform differently depending on the context.

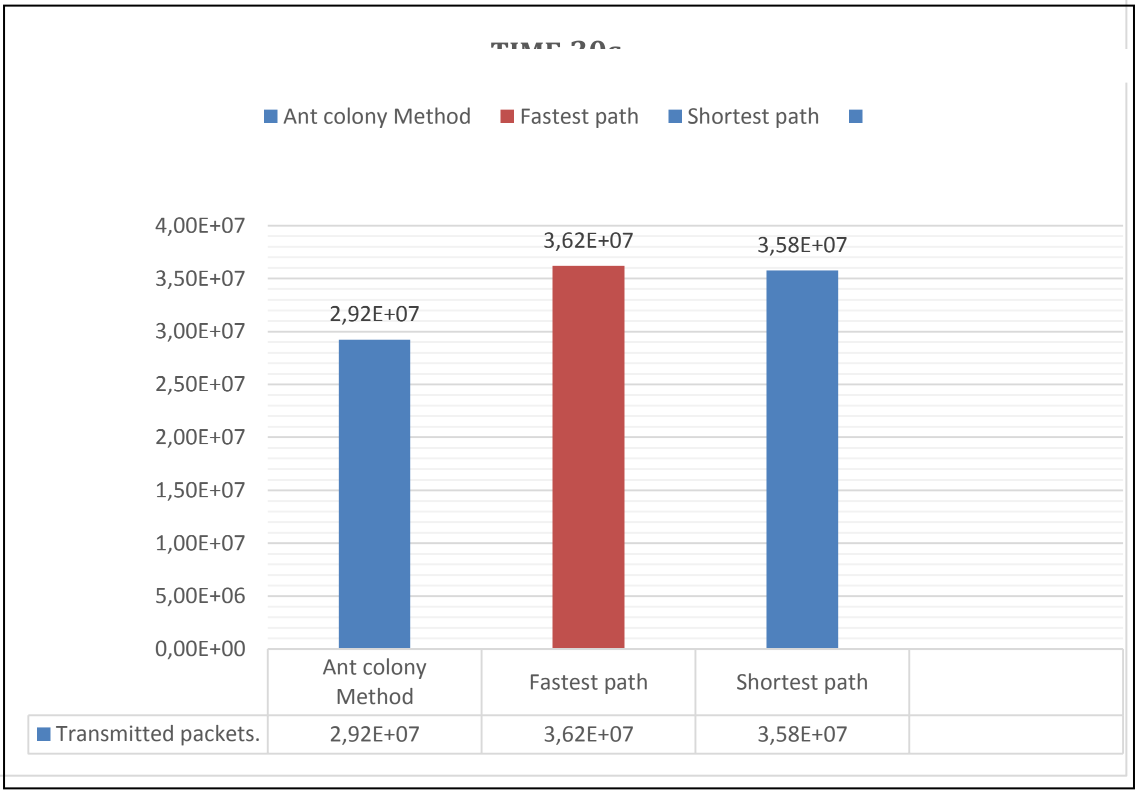

Figure 4.25: The Comparison Of The Throughout For Three Methods When Used Time 30s.

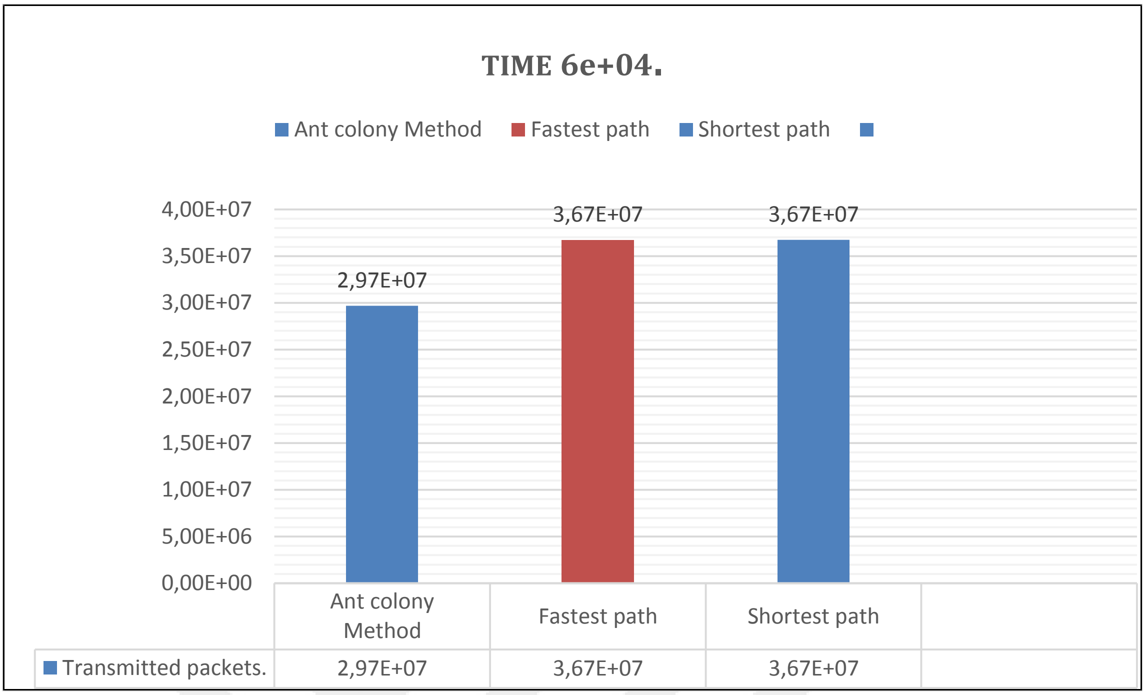

Figure 4.26: The Comparison Of The Throughout For Three Methods

When Used Time 6e+04.

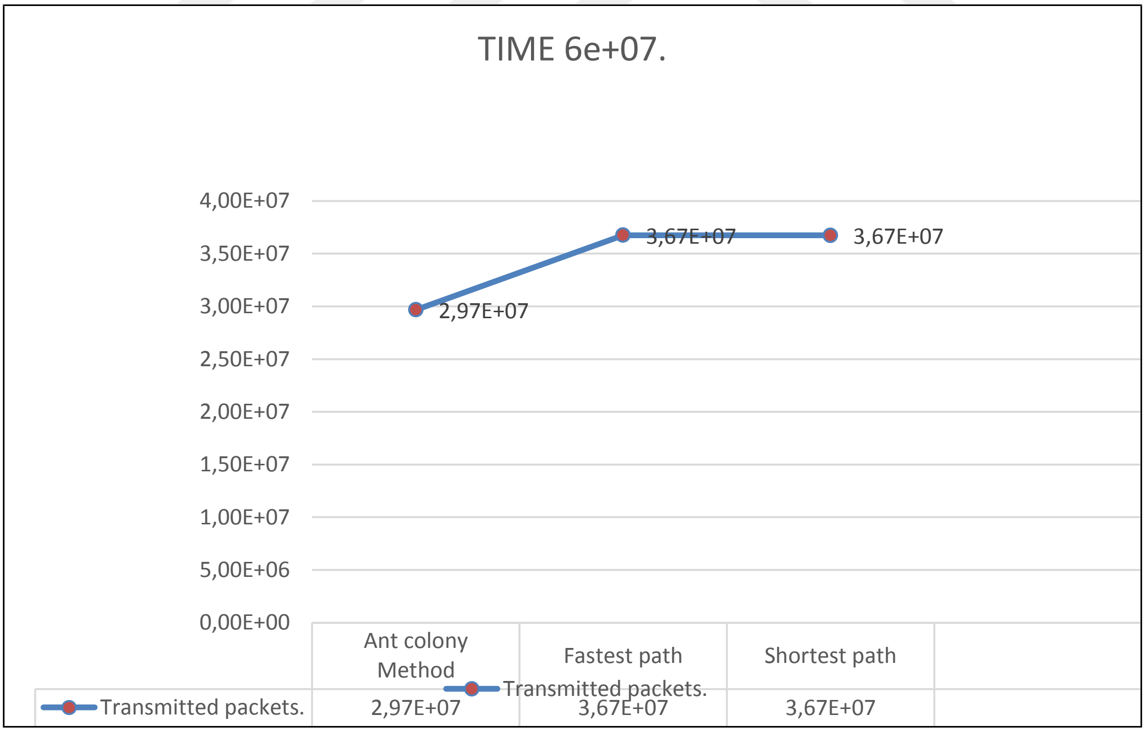

Figure 4.27: The Comparison Of The Throughout For Three Methods

When Used Time 6e+07.

4.4 CHAPTER SUMMARY

The results chapter discusses the three algorithms proposed in the study, namely Ant colony, shortest path, and fastest path, and their effectiveness in routing packets. The aim of the study is to evaluate the performance of these algorithms in terms of different metrics such as delay count, loss rate, average hop count, and number of packets sent per second. The simulation was conducted for both short-term and long-term implementations, and the performance of the algorithms was compared and analyzed. The study also explored novel cognitive routing techniques that can improve the transmission of voice and video packets.

The chapter further discusses the effects of different routing algorithms on packet throughput and how implementing recovery methods can improve network performance. The study compared the proposed algorithms with existing methods in a multi-level protocol network and demonstrated their ability to enhance network performance significantly.

Finally, the results of the simulation indicated that the proposed algorithms can provide a fast and simple way to route packets, and implementing recovery methods can improve network performance. The chapter concludes by highlighting the significance of the study's findings and their potential implications for future research in the field of networking.

5. CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

In conclusion, this research has analyzed and compared various algorithms for their throughput values. The study has demonstrated that cognitive methods can significantly enhance network throughput, and the proposed approaches, using either ant colony or the fastest path, can achieve desired throughput. Moreover, the findings have revealed that the fastest route provides a higher throughput value than the shorter path and ant colony.

Based on the analysis, the study concludes that the fastest path algorithm is the most efficient in improving network performance, while the shortest path algorithm is less efficient than the other two. The study's results provide valuable insights into the effectiveness of different algorithms in improving network throughput, which can help network designers and operators select the most appropriate algorithm for their specific requirements.

In summary, this research highlights the importance of cognitive methods and efficient algorithms in enhancing network performance. It provides practical guidance for selecting the most effective routing algorithm based on specific network requirements and highlights the potential for future research in this field.

5.2 FUTURE WORK

As a future work, the results of this study can be utilized to further enhance the performance of networks. One potential area of research is to compare and design other methods for the transmission of data within the multi-line transport network. This can help identify new algorithms or techniques that can further improve network performance.

Another recommendation is to analyze and compare several algorithms to determine which ones are more efficient in improving the network's performance. This can help provide new insights into the strengths and limitations of different routing algorithms and guide the selection of the most appropriate algorithm based on specific network requirements. Furthermore, future research can explore the impact of different parameters on network performance, such as the size of the network, the number of nodes, and the type of multimedia traffic. This can help develop a more comprehensive understanding of the factors that affect network performance and provide insights into the design and optimization of mobile multimedia networks. Overall, the findings of these future studies can be used to develop new networking solutions and improve our understanding of how to enhance the performance of mobile multimedia networks.

This can help meet the increasing demand for high-quality multimedia services in mobile networks and enable new applications and services that require high bandwidth and low latency.

REFERENCES

[1] D. Shumeye Lakew, U. Sa'ad, N. Dao, W. Na, and S. Cho, "Routing in flying ad hoc networks: A comprehensive survey," *IEEE Communi- cations Surveys Tutorials*, vol. 22, no. 2, pp. 1071–1120, 2020.

[2] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," CA. Cancer J. Clin., vol. 68, no. 6, pp. 394–424, 2018.

[3] W. H. Organization, WHO position paper on mammography screening. World Health Organization, 2014.

[4] A. Chetlen, J. Mack, and T. Chan, "Breast cancer screening controversies: Who, when, why, and how?, vol. 40, no. 2, pp. 279–282, 2016.

[5] A. Chekkoury et al., "Automated malignancy detection in breast histopathological images," in Medical Imaging 2012: Computer-Aided Diagnosis, 2012, vol. 8315, p. 831515.

[6] I. Pollanen, B. Braithwaite, T. Ikonen, H. Niska, K. Haataja, P. Toivanen, and T. Tolonen, "Computer-aided breast cancer histopathological diagnosis: Comparative analysis of three DTOCS-based features: SW-DTOCS, SW-WDTOCS and SW-3-4-DTOCS," 2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA), 2014.

[7] F. F. Ting, Y. J. Tan, and K. S. Sim, "Convolutional neural network improvement for breast cancer classification," Expert Syst. Appl., vol. 120, pp. 103–115, 2019.

[8] H. Ravishankar et al., "Understanding the mechanisms of deep transfer learning for medical images," arXiv Prepr. arXiv1704.06040, 2017.

[9] S. Lu, Z. Lu, and Y.-D. Zhang, "Pathological brain detection based on Alexnet and Transfer Learning," Journal of Computational Science, vol. 30, pp. 41–47, 2019.

[10] Y. Chen et al., "A feature-free 30-disease pathological brain detection system by linear regression classifier," CNS Neurol. Disord. Targets (Formerly Curr. Drug Targets-CNS Neurol. Disord., vol. 16, no. 1, pp. 5–10, 2017.

[11] S. Wang, P. Phillips, A. Liu, and S. Du, "Tea category identification using computer

vision and generalized eigenvalue proximal SVM," *Fundam. Informaticae,* vol. 151, no. 1–4, pp. 325–339, 2017.

[12] M. M. Rahman, Y. Ghasemi, E. Suley, Y. Zhou, S. Wang, and J. Rogers, "Machine Learning Based Computer Aided Diagnosis of Breast Cancer Utilizing Anthropometric and Clinical Features," Irbm, vol. 42, no. 4. pp. 215–226, doi: 10.1016/j.irbm.2020.05.005.2021.

[13] N. I. R. Yassin, S. Omran, E. M. F. El Houby, and H. Allam, "Machine learning techniques for breast cancer computer aided diagnosis using different image modalities: A systematic review," *Computer Methods and Programs in Biomedicine,* vol. 156. pp. 25–45, doi: 10.1016/j.cmpb. 12.012.2018.

[14] R. Yan et al., "Breast cancer histopathological image classification using a hybrid deep neural network," Methods, vol. 173. pp. 52–60, doi: 10.1016/j.ymeth. 06.014.2020.

[15] N. Harbeck et al., "Breast cancer," Nat. Rev. Dis. Prim., vol. 5, no. 1, pp. 1–31,

[16] A. G. Waks and E. P. Winer, "Breast cancer treatment: a review," *Jama, vol*. 321, no. 3, pp. 288–300, 2019.

[17] O. J. Scully, B.-H. Bay, G. Yip, and Y. Yu, "Breast cancer metastasis," *Cancer Genomics Proteomics*, vol. 9, no. 5, pp. 311–320, 2012.

[18] W. J. Allard et al., "Tumor cells circulate in the peripheral blood of all major carcinomas but not in healthy subjects or patients with nonmalignant diseases," *Clin. cancer Res*., vol. 10, no. 20, pp. 6897–6904, 2004.

[19] A. van de Stolpe, K. Pantel, S. Sleijfer, L. W. Terstappen, and J. M. J. Den Toonder, "Circulating tumor cell isolation and diagnostics: toward routine clinical use." *AACR*, 2011.

[20] N. Houssami and P. Skaane, "Overview of the evidence on digital breast tomosynthesis in breast cancer detection," *The Breast,* vol. 22, no. 2, pp. 101–108, 2013.

[21] S. W. Fu, L. Chen, and Y. Man, "miRNA biomarkers in breast cancer detection and management," *J. Cancer,* vol. 2, p. 116, 2011.

[22] S. G. Kandlikar et al., "Infrared imaging technology for breast cancer detection–Current status, protocols and new directions," *Int. J. Heat Mass Transf*., vol. 108, pp. 2303–2320, 2017.

[23] K. Ganesan, U. R. Acharya, C. K. Chua, L. C. Min, K. T. Abraham, and K.-H. Ng, "Computer-aided breast cancer detection using mammograms: a review," *IEEE Rev. Biomed. Eng.*, vol. 6, pp. 77–98, 2012.

[24] R. Guo, G. Lu, B. Qin, and B. Fei, "Ultrasound imaging technologies for breast cancer detection and management: a review," *Ultrasound Med*. Biol., vol. 44, no. 1, pp. 37–70, 2018.

[25] S. J. Mambou, P. Maresova, O. Krejcar, A. Selamat, and K. Kuca, "Breast cancer detection using infrared thermal imaging and a deep learning model," *Sensors*, vol. 18, no. 9, p. 2799, 2018.

[26] K. Suzuki, "A review of computer-aided diagnosis in thoracic and colonic imaging," *Quant. Imaging Med.* Surg., vol. 2, no. 3, p. 163, 2012.

[27] A. F. Agarap, "On breast cancer detection,"Proceedings of the 2nd International Conference on Machine Learning and Soft Computing - *ICMLSC* '18, 2018.

[28] M. Karabatak, "A new classifier for breast cancer detection based on Naïve Bayesian," Measurement, vol. 72, pp. 32–36, 2015.

[29] R. Mormont, P. Geurts, and R. Marée, "Comparison of deep transfer learning strategies for digital pathology," *in Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 2262–2271.2018.

[30] Y. Khourdifi and M. Bahaj, "Applying best machine learning algorithms for breast cancer prediction and classification," International Conference on Electronics, Control, Optimization and Computer Science, ICECOCS. doi: 10.1109/ICECOCS.2018.8610632.2018.

[31] P. Berkhin, "A survey of clustering data mining techniques," *in Grouping multidimensional data, Springer*, pp. 25–71.2006.

[32] C. I. Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, "A Survey on Deep Learning in Medical Image Analysis," *Med. Image Anal*, vol. 42, pp. 60–88, 2017.

[33] G. Litjens et al., "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, 2017.

[34] B. Mahesh, "Machine Learning Algorithms-A Review," *Int. J. Sci. Res.*

(IJSR).[Internet], vol. 9, pp. 381–386, 2020.

[35] A. Bhardwaj and A. Tiwari, “Breast cancer diagnosis using Genetically Optimized Neural Network model,” *Expert Systems with Applications,* vol. 42, no. 10. pp. 4611–4620, doi: 10.1016/j.eswa..01.065.2015.

[36] N. Khuriwal and N. Mishra, “Breast cancer diagnosis using deep learning algorithm,” *in International Conference on Advances in Computing, Communication Control and Networking (ICACCCN*), pp. 98–103.2018.

[37] Y. Yu, H. Lin, J. Meng, X. Wei, H. Guo, and Z. Zhao, “Deep Transfer Learning for Modality Classification of Medical Images,” doi: 10.3390/info8030091.2017.

[38] P. Gupta and S. Garg, “Breast Cancer Prediction using varying Parameters of Machine Learning Models,” Procedia Computer Science, vol. 171. pp. 593–601, doi: 10.1016/j.procs. 04.064.2020.

[39] L. Zou, S. Yu, T. Meng, Z. Zhang, X. Liang, and Y. Xie, “A technical review of convolutional neural network-based mammographic breast cancer diagnosis,” *Comput. Math. Methods Med.*, vol. 2019.

[40] A. Titoriya and S. Sachdeva, “Breast Cancer Histopathology Image Classification using AlexNet,” *4th International Conference on Information Systems and Computer Networks,* ISCON. pp. 708–712, doi: 10.1109/ISCON47742.2019.9036160.2019.

[41] Dong y., Gu J., Li N., Hou X., Yan W., (2007), “Combination of Genetic Algorithm and Ant Colony Algorithm for Distribution Network Planning”, Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Tianjin, China, pp. 1-4

[42] Xu Y., Wang W., (2009), “Finding the Fastest Path in Wireless Networks”, North Carolina State University, USA, pp. 1-5.

[43] Nagib G. and Wahied G. A., (2010), *“Network Routing using Genetic Algorithms”,* International Journal of Electrical and Computer Sciences, vol. 10, no. 2, pp. 1-5.