Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Performance Evaluation of Optimization Algorithms for BGP Slow Convergence Problem

By Mihret Gebre

Advisor Dr. Yalemzewd Negash

A Thesis Submitted to the School of Electrical and Computer Engineering of Addis Ababa Institute of Technology in Partial Fulfillment of the Requirements for the Degree of Master of Science in Telecommunication Engineering

January 2022

Addis Ababa, Ethiopia

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Telecommunication Engineering Graduate Program

Performance Evaluation of Optimization Algorithms for BGP Slow Convergence Problem

By Mihret Gebre

Approval by Board of Examiners

______________________	______________	______________
Chairman / School Dean	Signature	Date
Dr. Yalemzewd Negash	______________	______________
Advisor	Signature	Date
Dr. Yihenew Wondie	______________	______________
Examiner	Signature	Date
Dr.Murad Ridwan	______________	______________
Examiner	Signature	Date
______________________	______________	______________
Director, Postgraduate Program	Signature	Date

DECLARATION OF ORIGINALITY

I, the undersigned, declare that this thesis and the work presented in it are my own and have been generated by me as the result of my original research. I truly acknowledged and referred to every material which used in this thesis work.

MIHRET GEBRE ________________________

Name Signature

Addis Ababa, Ethiopia ________________________

Place Date of submission

This thesis has been submitted for examination with my approval as a university advisor.

Dr. Yalemzewd Negash ____________________

Advisor's Name Signature

DEDICATION

This thesis is dedicated to my Golden family for their limitless love, endless support, and encouragement.

And

I also dedicate this work to all women who raise their child/ren lonely.

ABSTRACT

Border Gateway Protocol (BGP) is an inter-domain routing protocol that provides routing or reachability information inside or between different Autonomous Systems (AS). The existing inter-domain routing architecture has a major challenge due to slow convergence during network failures. The slow routing convergence time results in intermittent loss of connectivity, increased packet loss, and latency. The Minimum Route Advertisement Interval (MRAI) timer limits the number of messages propagated by BGP speakers. Convergence time can be characterized in terms of MRAI rounds. Thus, the optimum MRAI timer implementation plays a vital role in improving the convergence time of BGP.

This thesis is conducted to evaluate and analyze the performance of optimization algorithms to determine an optimum value for the MRAI timer, which minimizes the convergence time without affecting the number of update messages. The dataset has been gathered from the network topology with different values of MRAI using Graphical Network Simulator-3 (GNS3). The Adaptive Neuro-Fuzzy Inference Systems (ANFIS) is trained with the dataset to generate a model. Finally, three optimization algorithms such as Artificial Bee Colony (ABC), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) are applied to the model to get the optimum value for the MRAI timer. The implementation has been performed using MATLAB.

The results indicated that the PSO model outperforms the ABC and GA which reduces the running time and lowers the convergence rate required to complete the search process. The experimental results and analysis also show that in optimized BGP convergence time is improved by 55%, and packet loss is improved by 19% as compared to default BGP.

Keywords: BGP, Convergence time, MRAI, ANFIS, Optimization Algorithm, Performance.

ACKNOWLEDGMENTS

First and foremost, I would like to express my special thanks to my almighty God for providing me with the courage and endurance to successfully complete my thesis. I am so grateful to my company Ethio telecom and Electrical and Computer Engineering School at Addis Ababa University, for giving me this opportunity.

Next, my deep gratitude goes to my advisor, Yalemzewd Negash (Ph.D.) for his valuable and polite supervision that has been done during the entire journey of the thesis. It is a great chance to work on this thesis with someone who always offered generously his time and knowledge whenever I am in need of advice. Besides my advisor, I would also like to thank my thesis examiners Yihenew Wondie (Ph.D.), and Murad Ridwan (Ph.D.), for their suggestions and comments which were very valuable during various phases of the thesis completion.

In addition, I would like to thank my classmates and childhood friends for their encouragement and patience. And my sincerest thanks to Zafu Fissiha, my amazing sister, for your prayers and for being there for Natu whenever needed.

At last, but not least, I wish to take this opportunity to acknowledge my lovely parents and my second mother Birha for their limitless love, support, and guidance. My world Natu, this journey is intended to teach you the strength and endurance that you need throughout your life. To the twins, who are the most incredible and generous brothers and who are always there for me no matter when or where. I am so grateful for everything you have done for me. Thank you with all my heart.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
AI	Artificial Intelligence
ANFIS	Adaptive Neuro-Fuzzy Inference Systems
AS	Autonomous System
ASN	Autonomous System Number
BGP	Border Gateway Protocol
BR	Border Routers
CE	Customer Equipment
CPU	Central Processing Unit
CR	Core Routers
DoP	Degree of Preference
eBGP	external BGP
ER	Edge Routers
FIB	Forwarding Information Base
GA	Genetic Algorithm
GNS3	Graphical Network Simulator-3
GUI	Graphical User Interface
iBGP	internal BGP
IGP	Interior Gateway Protocol
IP	Internet Protocol
ISP	Internet Service Provider
IS-IS	Intermediate System-to-Intermediate System
MF	Membership Function
MPLS	Multi-Protocol Label Switching
MRAI	Minimum Route Advertisement Interval
MSE	Mean Square Error
OS	Operating System
PE	Provides Edge
PSO	Particle Swarm Optimization
QoS	Quality of Service

RAM	Random-Access Memory
RCN	Root Cause Notification
RFD	Route Flap Damping
RIB	Routing Information Base
RID	Router ID
RMSE	Root Mean Square Error
RR	Route Reflector
RTT	Round Trip Time
SDN	Software-Defined Networking
SI	Swarm Intelligence
VRF	Virtual Routing and Forwarding
VRRP	Virtual Router Redundancy Protocol
VPN	Virtual Private Networks

1

INTRODUCTION

This chapter sets out the background to this research and explains the statement of the problem, objectives, scope, contribution of the research. In addition, it explains the approach and methodology used as well as reviewed literatures related to this study are discussed. In the last section of this chapter, the outline of the thesis is also illustrated.

1.1 BACKGROUND

The Internet is composed of different separately regulated network domains known as Autonomous System (AS). Every AS has a unique identifier called an Autonomous System Number (ASN). Border Gateway Protocol (BGP) executes routing on the AS network [1]. ASN with BGP permits the exchange of routing information between its AS and other systems.

BGP is a path-vector routing protocol mainly performed on routers placed at the topological boundary of the Autonomous Systems (ASes), and it contains a set of metrics for the election of the best path close to the destination. The BGP speaker is a router in which BGP is running, and BGP peers or neighbors are BGP speakers that are establishing a BGP connection while exchanging routing information [2]. BGP sessions are established between BGP neighbors to interchange routing updates with in the same or different ASes. BGP speakers are classified as internal and external according to the level of connection they establish. The internal BGP (iBGP) speaker allows the routers to exchange routes within the same AS whereas the external BGP (eBGP) speaker provides routes between neighbors in different ASes [1]. To exchange routing information and updates between iBGP peering, the iBGP architecture falls into three designs: full-mesh, AS confederation, and Route-Reflector (RR) [3].

In the full-mesh configuration, all BGP routers within the same AS are directly connected through pairwise iBGP sessions; this will enable each BGP router to know about reachability information directly from all other BGP routers within the same AS to eliminate routing loops [3]. In the second configuration, it forms various sub-ASes inside AS by dividing a large BGP AS into smaller ASes, and the whole groups are delegated to a single confederation so that there will be a smaller number of iBGP routers in every sub-AS; this results in a smaller number of iBGP sessions inside sub-AS. Communication with other confederations and each sub-ASes link up with each other inside the AS is done using eBGP [3]. In the RR architectural connectivity, one or two routers are considered as a route reflector, and the remaining routers in AS established session with RR. The BGP update messages are accepted by RR from every iBGP speaker and propagated or reflected to all other iBGP speakers [4].

The total number of iBGP sessions increased as the square of the number of iBGP routers within the network, due to this the full mesh faces the scaling problem [5]. AS confederation still requires full iBGP within each sub-AS, which offers medium scalability as compared to RR. Even though route reflector provides high scalability, reduces the number of iBGP sessions and operational cost, however, it introduces prolonged routing convergence [3].

Convergence time is an important factor in defining routing protocol performance, and routing protocol is a key aspect in defining the quality of Internet Protocol (IP) communication. It is a period needed by the routing protocols to re-transmit the packet upon the failure or routing change. As a result, a network's convergence time is critical, and networks that converge quickly are seen as more stable [6,7]. According to [3] during network outages, the existing inter-domain routing architecture faces a significant challenge which is slow convergence. This problem was addressed by different approaches such as using redundant standby paths to ensure the performance of the data plane during routing convergence, minimizing Minimum Route Advertisement Interval (MRAI) timer to accelerate routing propagation, and designing effective route flap damping to prevent update flooding with minimized MRAI time value. The convergence characteristic also related to the relationship between the message handling process, the MRAI timer, and the network topology [7].

The MRAI timer is used in iBGP and eBGP sessions to limits the number of messages propagated by BGP speakers. The BGP speaker announces updates as the MRAI timer expires. Moreover, MRAI plays an essential role in BGP since the convergence time can be characterized in terms of MRAI rounds [7,8]. The default value of MRAI is 30 and 5 seconds for eBGP and iBGP respectively [9].

For some time intervals, the MRAI timer keeps the updates. In the meantime, the router chooses the best route to a destination and exchanges with other routers fewer updates. MRAI timer gets greater convergence in this manner, but MRAI timer can also worsen the convergence if changes are kept excessively [4,10]. Thus, the optimum MRAI timer implementation plays a vital role in improving the convergence time of BGP.

Therefore, it is desirable to investigate the optimum value of MRAI to enhance the convergence time of BGP. The proposed approach has been validated using three optimization algorithms, such as Artificial Bee Colony (ABC), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) for slow convergence problem that aims to find the optimum value of MRAI which reduces the BGP convergence delay, limit the effect of large-scale Internet failures as well as improve network performance.

1.2 STATEMENTS OF THE PROBLEM

Ethio telecom backbone network deploys BGP inter-AS routing protocol to interchange a large number of route entries with international Internet Service Providers (ISPs).

Since the gateway devices are connected to other networks (outside Ethiopia) directly, they need to deploy eBGP. Ethio telecom service bearer network only requires all devices deployed in the network to keep detailed routes of overall network services, so iBGP is applied. Ethio telecom also deployed two Route Reflectors for iBGP implementation, redundancy as mutually backups, which are responsible for route convergence, forwarding as well as reflecting Virtual Private Network version 4 (VPNv4) route [11].

The routing protocol design considers the convergence time as an important factor for routing efficiency. An efficient routing table and a faster convergence time are necessary when adding or withdrawing network elements. However, slow convergence issues have been among the most serious challenges for service providers in general and Ethio telecom in particular. Due to network instability, topology change, and routing policy, BGP brings high convergence time which is in the range of three minutes to fifteen minutes. During this time, the data traffic will experience service unavailability, increased packet loss, cause additional overhead to routers and long round-trip delays [6], which degrade the quality of service and overall network performances.

As the MRAI increases, nodes must wait longer before additional updates are sent which will prolong the delay in convergence. On the other hand, if the MRAI timer is smaller, updates are produced faster and the delay decreases but the router Central Processing Unit (CPU) loads increases [8]. Hence, finding the optimum value for MRAI timers is necessary to address the problem of BGP slow convergence.

This study provides a solution that contributes to overcoming the BGP slow convergence problem using optimization techniques. This optimization process will be applied to the dataset, gathered from the existing network topology, to determine the optimum value of the MRAI timer. Finally, this research investigates and seeks answers to the following research questions.

1. Which optimization technique can be more effective and achieve better performance to solve BGP slow convergence problem?
2. Which MRAI value is optimum for better network performance?

1.3 OBJECTIVES OF THE RESEARCH

1.3.1. General Objective

The general objective of this study is to compare the performance of three optimization algorithms (Artificial Bee Colony, Genetic Algorithm, and Particle Swarm Optimization) to determine the optimum value for MRAI timer in iBGP Route-Reflector architecture.

1.3.2. Specific Objective

The specific objective of this research includes:

- To analyze the impact of the iBGP deployment design on the BGP convergence time.
- To investigate the reduction of BGP convergence time by using the MRAI timer.
- To identify algorithms and tools for optimization of MRAI value.
- To generate and capture the dataset from the network topology.
- To test and evaluate the performance of selected algorithms.
- To identify the optimum value for MRAI.
- To compare the network performance using default and optimum MRAI timer.
- To provide a proper recommendation based on the results.

1.4. SCOPE

This thesis work mainly considers the Addis Ababa Microwave and Bole site, which uses RR configuration, of Ethio telecom bearer network architecture. Besides this, the research compares three optimization algorithms in which each algorithm uses different parameters. These parameters in each algorithm are selected based on the available research recommendations for their best and efficient results.

1.5. SIGNIFICANCE OF THE STUDY

This study will help Ethio telecom to enhance the network's performance by improving the convergence time of BGP. This will speed up the rerouting of the packets, reducing packet loss and delay in the event of failure, enhancing the quality of service for the company's customers and overall network performance.

1.6. RELATED WORKS

Here, literatures are collected, studied, and analyzed to select the most relevant among them. Google books, IEEE Explore, Research Gate, and vendor websites are the main sources of the literature selected. Related potential literatures are also selected, examined, compared, and evaluated for the soundness of this proposal.

The authors in [1] provided a more exhaustive literature review on the inter-domain routing performance improvement methods. The concept of Internet convergence time is the period taken to propagate the network topology change due to the addition or withdrawal of a node throughout the network. The authors also examined how the delay in Internet convergence has been measured. There are two approaches to measure routing convergence on the Internet. The first option uses the Internet public databases containing information about routes of several places around the world. In contrast, the second alternative uses topology generators to simulate part of the Internet and generate an adjustment to BGP protocol, then examining the impact of the changes on convergence time. Among the main problems that affect inter-domain routing convergence are poor understanding of BGP behavior, policy conflicts, momentary failures and oscillations, ghost information, and extended table size and path exploration. Finally, how the slow convergence problem was addressed by the different approaches such as speeding up, limiting path exploration, efficient policy configuration, multi-path, and multi-path forwarding, and centralized control are discussed. The authors didn't add new mechanisms, they only contribute to the growth and advancement of the reduction of inter-domain convergence delay research.

To ensure the high scalability of the network, BGP must eliminate the instability. Instability and failure will lead the network to an unstable state, which greatly increases the convergence time of the network. The current efforts to improve the convergence speed of the BGP protocol such as BGP policies, instability, and fault detection have been summarized in [12].

The effect of the MRAI timer on the network convergence time and the number of update messages received are studied by [13]. The authors implemented different values of MRAI to examine the impact of MRAI on the convergence time and found that if the MRAI timer increases, convergence time will increase, but the number of updates decreases. This approach does not provide any optimal value of MRAI for enhancing the convergence time and the number of updates.

The slow convergence problem can be reduced by using the optimum value of MRAI. However, only the updates are propagated in the network when it is set to zero. The authors in [10] proposed the Fickle Mean Route Advertisement Interval (FMRAI) technique to improve the time of network convergence. The suggested algorithm enhances a lower packet drop ratio, the convergence time, and the number of updates relative to the default MRAI value. However, for enhancing the convergence time, due to their high performance and simplicity, the approach should implement Swarm Intelligence (SI) optimization techniques that identify the global minimizer to resolve the issue of fast convergence time.

The authors of [14], studied the optimization of MRAI to reduce BGP convergence time. Various network topologies from the Internet are generated for network graphs. For each network graph, various values of MRAI have been utilized. The result has been recorded and used as a dataset to train the Neuro-Fuzzy system. Then PSO is implemented to determine the optimal value of MRAI. The solution suggests a value of

3 seconds for MRAI, but the recommended value is not suitable for heavy traffic load variations.

The authors in [4] studied improving convergence in iBGP Route Reflectors using an adjustable MRAI timer. Adjustable MRAI timers can limit the updates of different destinations. The timer holds a queue. The number of prefixes that the peers advertise is called prefix load. When the MRAI ends, the router deletes the prefix from the queue. The algorithm uses the adjustable MRAI timer when the prefix load is greater than the upper bound otherwise, the default MRAI is used. For iBGP route reflectors, the adjustable MRAI timer achieves improved convergence. But this technique greatly increases the number of exchanged updates as well as the router overheads.

A new optimized MRAI method is proposed by [15] to determine an optimal number of MRAI timers under the high traffic load. Optimized MRAI determines the required number of MRAI timers and granularity based on the Granularity Identifier Function (GIF). Optimized MRAI dynamically initializes granularity and the required number of MRAI timers in each MRAI round. The suggested approach aims to reduce the BGP convergence time, end-to-end delay, the number of exchanged update messages, and enhance the number of delivered packets relative to the Adaptive MRAI and Flexible MRAI approach.

1.7. METHODOLOGY

In this section, the methodology to be followed to carry out this thesis is discussed. To address the stated problems and also to achieve the goals that are mentioned in general as well as in specific objectives, the following methods were followed:

1. Conducting a systematic literature review to better understand the research area, such as a review on existing iBGP deployment designs, the MRAI timer's impact on convergence time as well as investigate the mentioned iBGP deployment architecture problems in depth. Different books, journals, articles, and papers

from the Internet were reviewed to determine the importance and implementations of the BGP convergence delay problem with optimization algorithm approaches.

2. Select the appropriate and well-known Intelligent modeling system and optimization algorithms.
3. The existing network topology, which implements BGP on RR, is configured using GNS3. Then the data will be generated from the convergence time to train the modeling system.
4. Applying optimization algorithms on the model from the system to compare, discuss, and analyze the performance of the optimization algorithms. Based on the comparison result the optimum value of MRAI will be selected from the best algorithm.
5. In the subsequent step, the test results for the two scenarios, such as BGP with default MRAI timer and BGP with optimum value of MRAI, will be presented graphically for performance comparison of the network.
6. Finally, conclusions and recommendations will be drawn based on the findings.

Based on the above steps the flow chart of the methodology employed in this study is illustrated in the figure below.

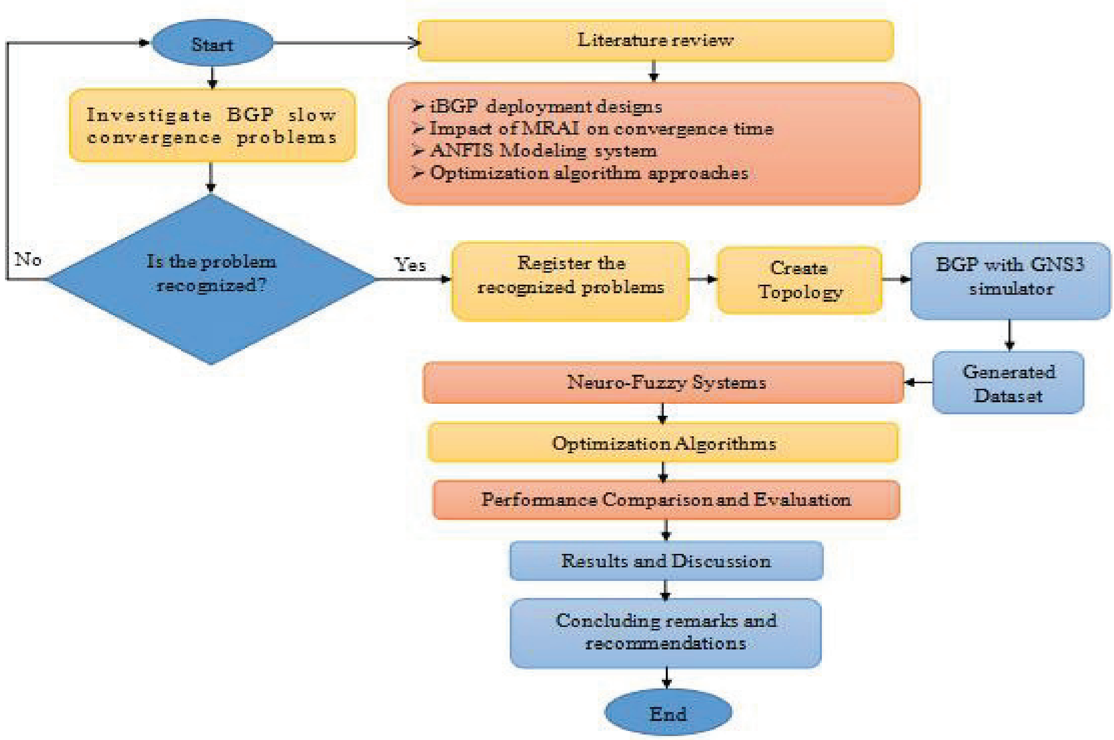


Figure 1: Research methodology flow chart

1.8. RESEARCH ORGANIZATION

This research work is structured into six chapters and organized as follows. Chapter 2 focuses on the literature review to provide a fundamental understanding of Border Gateway Protocol, the main issues on inter-domain routing convergence, and provides efforts done in reducing convergence delay. Chapter 3 introduces basic concepts in Adaptive Neuro-Fuzzy Inference Systems (ANFIS) and optimization algorithms that are used in this study. Chapter 4 deals with BGP simulation, data generation process, model experimentation, model performance as well as network performance evaluation metrics. Chapter 5 focuses on experimental results and analysis of the outcomes of the experiment. Finally, Chapter 6 provides the conclusion of the study and suggests recommendations for further work.

2

LITERATURE REVIEW

This chapter presents the state-of-the-art and literature review related work for this research. It provides an overview of the background about Border Gateway Protocol, its states and messages, route processing, and routing convergence delay followed by a description of major approaches to reduce inter-domain routing convergence delay on the Internet.

2.1. BORDER GATEWAY PROTOCOL (BGP)

Internet routing protocols are split into two categories such as intra-domain and inter-domain protocols. Intra-domain or Interior Gateway Protocols (IGP) are used within ASes for routing [16]. Whereas Inter-domain routing protocols are used to route data between different AS. One of the most important and standard inter-domain routing protocols on the Internet is the BGP [9]. It is the de facto standard inter-domain routing protocol on the Internet that links tens of thousands of networks on the Internet to create one huge network that's interconnected. BGP shares information about network reachability between the AS domain, generally in the border routers. The AS is a collection of routers under a single technical administration [17].

2.1.1. BGP states and messages

To start communication between BGP peers, BGP defines the method of creating a BGP session. The BGP session can be in Idle, Connect, Active, Open Sent, Open Confirm, or Established state. The router does not set up a BGP session in the idle state and waits for the BGP start event. By listening to the incoming Transmission Control Protocol (TCP) session in the connected state, the router establishes the TCP connection between the BGP peers. In the active state, BGP waits for a TCP session from its neighbor. To complete a BGP session, the open sent and open confirmed states are used. A connection is ready for transmission of updates, keepalive, and notification messages after confirmations of open sent and open confirm states [17].

On the other hand, in order to run BGP, BGP routers should form a neighbor relationship with adjacent BGP speakers. Once BGP peers establish a neighbor relationship, they share their routing information. After that, only updates to the routing table are forwarded to peers. The routing information transmitted consists of the complete route to the desired destination. BGP uses this routing information to maintain a network reachability information database, that is shared through peer-to-peer communication via BGP Speaker with other BGP systems or neighbors [18]. If two BGP peers are connected, four distinct types of messages may be exchanged to each other: OPEN, KEEPALIVE, UPDATE, and NOTIFICATION [1]. However, a TCP link between the two BGP peers is required for a connection to be formed first. BGP uses TCP on port 179 for reliable packet transfers. The OPEN message is the first message that the BGP speakers have shared. If an error is not detected in this message, the session is initialized and essential information is learned about each speaker. Each speaker will send an UPDATE message in the next step. The speaker will download the whole routing table from its neighbor when this message is first received. This message can involve removal to a route that no longer exists or a new best path announcement. When any errors are occurred in a BGP message or during session establishment among BGP peers, the peer that identified the error, resulting in at the end of the session, sends a NOTIFICATION message. To ensure that the session persists, the KEEPALIVE messages are exchanged at a predetermined interval. In addition to these, BGP has a timer that limits event notification which is known as the MRAI. Before transmitting new updates, an update message should wait for some time equal to the MRAI value [16,19, 20].

2.1.2. Route processing in BGP

There are network advertisements (for new and existing routes) and withdrawals in the update message. When a router chooses a path to a new network, it sends network advertisements to its peers. When a router can't forward data via the selected path to a network and no alternative path is available for that network, network withdrawals are sent to its peers. There are a set of BGP attributes for selecting the best path close to the destination in a network advertisement connected to the advertising network [21,22]. For route advertisement which can be used to choose the best route, several metrics such as Local preference, AS-path, Origin Type, Multi-exit Discriminator (MED) are added [23]. The Origin attribute is used by AS, to provide routing information to the BGP speaker. It is used for all the update messages to give information about the AS that issued it. AS-path involves details in AS sets as well as the sequences of ASes that the route is passed through. This attribute keeps changing with changes to the routes or new routes and the unavailability of previously announced routes [17]. MED attribute includes the inter-AS router information and allows them to connect with neighboring ASes. It enables routers to determine which routers will exchange packets for the destination AS. Local preference allows the desired route to be determined. For internal routing inside an AS, this attribute can only be added to a route [17]. BGP routing implements best path selection algorithms such as User-configured policies, highest local preference, shortest AS path, and lowest MED by prioritizing routes according to the Degree of Preference (DoP) [17]. To prevent routing loops, a router rejects any advertisement with a path vector that includes the router's ASN [2].

2.1.3. BGP convergence delay

In case of topological shifts due to link/node failure or policy changes, the routers might change their state from one correct and stable routing table condition to another correct and stable state by exchanging routing updates. This process is called convergence, and also the time taken by this process is named convergence time [24]. One reason why the Internet isn't so available is the convergence delay [1,25]. Different events can affect the BGP convergence process. There are two types of events in terms of topological changes, such as failure or recovery. Failure causes certain nodes/links to be inaccessible, and recovery returns some nodes/links to the system. Furthermore, failures can have two effects on the destination's reachability such as failover and fail down. In fail down, the destination from the network is disconnected by failure whereas in failover the nodes are forced to use an alternative path to reach the destination [22, 24].

2.1.3.1. Route Reflector convergence time

There is a simple relation between two nodes in an iBGP fully mesh topology. When there are changes in the networks, the updates are immediately obtained by each other. A route-reflector will provide longer routing convergence compared to full-mesh iBGP connections. Clients of RR and RR’s link with each other in a hierarchical manner and clients of route reflectors receive an update via route reflector. The number of hops increases due to the hierarchical topology of the route reflector. Besides that, the number of paths for path selection increases, and if any change exists in the topology, the number of updates quickly increases [3, 4]. In compliance with the route selection policy, a route reflector chooses a path to a destination and substitutes the existing route in a Routing Information Base (RIB). Furthermore, the exchange of information about changed routes frequently degrades the performance of the network [4]. Frequent changes in traffic loads have a more significant effect on BGP convergence time on large networks. The convergence of a system also relies on the availability of

BGP's interdomain routing service. The difficulties in configuration or understanding of BGP dynamics cause service unavailability. Moreover, several factors influence BGP's convergence time, such as the MRAI timer, the number of update messages interchanged between BGP routers, adopted topologies, timers, and overhead of routers during recovery. On the other hand, different causes, such as TCP disconnection, security problems, and malfunction incidents, will make the interdomain service inaccessible [26]. Improving convergence can help to re-route the packets more efficiently, reduce packet loss and delay during failure [4,15,19].

2.1.4. Minimum Route Advertisement Interval (MRAI) Timer

A BGP router divides the transmission of successive advertisements to the peer for a certain network; it uses MRAI to limit the number of update messages [2]. During convergence, the MRAI time attempts to speed up convergence toward bad behavior, which means transient routing, forwarding changes, and high overheads for routers. BGP speaker checks if the MRAI is not expired for that peer until it begins sending updates to any peer [19]. For a certain amount of time, the MRAI timer keeps the updates. In the meantime, the router chooses the best route to the destination and shares fewer updates. The duration and implementation of MRAI timers would have an important effect on the BGP convergence time. Therefore, the implementation of the MRAI timer is very important to enhance the convergence process [4,16].

The effect of MRAI value on convergence delay will be illustrated below [8]:

As the MRAI increases, nodes must wait longer before additional updates are sent, which will prolong the convergence delay. The number of updates sent in this process remains constant, and the time delay linearly increases.

If MRAI is smaller, updates are produced faster, but the router Central Processing Unit (CPU) load increases. So a node will give a neighbor an update before all update messages are processed. When one of the other updates changes the routes, it has just been advertised on the queue, and another update must be forwarded. The workload on nodes will be increased, which will, in turn, increase the delay in convergence.

2.2. PERFORMANCE PARAMETERS OF BGP

This section will provide the performance metrics of BGP such as link bandwidth, CPU utilization, memory requirement, and convergence time will be presented below [21].

Link Bandwidth: If the total number of routes on the Internet is given in M and the total path attributes obtained from a peer as A and assuming that the networks are spread equally between the autonomous systems, during the initial communication between a pair of BGP speakers the worst case is the amount of bandwidth used is [21]:

$$BW = O\,(M + A) \quad \text{(2. 1)}$$

Where: O is the order operator.

CPU Utilization: The CPU utilization by BGP relies solely on the stability of the network which relates to BGP in terms of BGP UPDATE message announcements. A uniformly distributed random variable from the interval $[P_{min}, P_{max}]$ is used to estimate the processing delay of each message. For a group of L queued updates, the average processing delay $T_{BGP\ process}(p)$ is [21]:

$$T_{BGPproces} = L \times \left[\frac{P_{max} - P_{min}}{2} \right] \quad \text{(2. 2)}$$

Memory Requirements: If K denotes the total number of networks on the Internet, P represents the BGP peers per network, the total number of unique AS paths as U, then memory requirements (MR) can be expressed as [21]:

$$MR = O\,((K + U) \times P) \quad \text{(2.3)}$$

Where: O is the order operator.

Convergence time: the convergence time on the lower bound is given by [13,21]:

$$\textit{Minimum convergence time} = (N\text{-}3) \times MRAI \quad \text{(2.4)}$$

Where: N is the number of nodes in a network

The maximum time of convergence that can be delayed by a route advertisement in one BGP speaker is equal to MRAI, and thus the upper limit for $t(p)$ is the product of MRAI and the maximum number of hops (H) as given in [13,21]:

$$Maximum\ convergence\ time = H \times MRAI \quad \text{(2.5)}$$

$$Average\ convergence\ time = \frac{(N\text{-}3 + H) \times MRAI}{2} \quad \text{(2.6)}$$

2.3. CONVERGENCE TIME REDUCTION APPROACHES

Reducing the Internet's convergence delay and message complexity provides many advantages, such as providing QoS and high availability of Internet services. The different approaches and classifications of efforts to address the slow convergence problem will be discussed in this section.

2.3.1. Descriptive and Analytical Approaches

Reducing the convergence time, in general, might also enhance the delivery of the packet. Since a long MRAI timer can considerably delay the system's convergence, re-examining its values may be an essential step in enhancing packet delivery. There are different solutions which are categorized in these approaches such as MRAI reconsideration provides a solution by checking BGP convergence with a failure followed by recovery; the MRAI value is re-examined [1]; MRAI Trade-off studies the effect of the MRAI value change by observing the compromise between updates and reduced router load. The MRAI impact analysis suggest several strategies for evaluating the effect on BGP-convergence behavior with different MRAI values [8,13,14].

2.3.2. Speeding up

Mainly this approach allows convergence faster by speeding update propagation, adjusting timer settings, or altering the selection process for the BGP path. Some of the strategies classified in this stream are ghost flushing and flight or fight. The ghost flushing approach reduces the delay of convergence from *n*30* to *d*h* by sending withdrawals regardless of hold-timer interruption. Where *d* is the length of the longer AS path, *n* is the number of nodes, and *h* is the average delay between two BGP neighbor routers [1]. The flight or fight approach speed up path exploration by discarding several path selection metrics when an update is a withdrawal [23].

2.3.3. Limiting path exploration

This strategy makes fewer network updates to circulate in a network, aiming to remove inconsistency and reducing the delay in convergence. Among the solutions that are included in this approach are: consistency assertions, Route Flap Damping (RFD) as well as Root Cause Notification (RCN). The first approach uses path information to compare paths to define inconsistent paths. The second one detects and suppresses unstable routes by setting penalties until the routing situation becomes stable. The last approach includes each AS listed in an AS-PATH with a sequence number indicating whether the route is valid or not. If a node detects a failure, its sequence number is updated and invalidating invalid alternative paths [1,19].

2.3.4. Efficient Policy Configuration

In this method, policy conflicts are resolved, and inconsistencies that cause network delays are mitigated. Some of the solutions listed in this category are: Dynamic oscillation detection, Policy combination and Policy analysis. The Dynamic oscillation detection identifies and addresses BGP oscillations based on the local policy. Policy combination recommends the combination of policy with the elimination of unstable routes to deal with oscillation. In the Policy analysis method, the impact of preference policies on convergence time is examined [1,19].

2.3.5. Multi-path and Multi-path forwarding

To prevent errors, multi-path identifies alternate routes to be employed when convergence events are identified without wasting time exploring RIBs searching for a new best route. In addition to the best ones, packets hold alternate paths, and when convergence events arise, they switch quickly to the alternative routes or may be routed to multiple next-hops. On the other hand, multi-path forwarding categorizes the approaches that send packets through several next-hops to the same destination. One of the objectives is to maximize the network bandwidth by using multiple paths to the destination's unused capacity. Some of the solutions listed in this approach are indicative re-routing scheme, flexible path as well as multi-path interdomain routing. In the first approach a simple indicator may be used to indicate the existence of alternate routes. Flexible path describes the advantages of versatile traffic segregation over many routes and multi-path interdomain routing controls traffic transit areas and guarantees high flexibility [1,19].

2.3.6. Centralized control

This approach enables high standards of powerful network management with the ability to develop new solutions to minimize Internet routing convergence. One of the most known approaches in this category is Software Defined Networking (SDN) which splits the network into two parts: the control plane, where logical decisions take place, and the data plane, where network equipment transmits traffic. There are main possible strategies to separate routing from routers such as routing control platform, routing as a service, virtual routers as a service, cloud-assisted routing, routing control platforms with a software-defined network, routing module in a controller, and SoftRouter. The routing issues that arise in SDN may fall in the control plane since any threat to the control plane has contributed to SDN architecture's failure [1,19,27].

3

ADAPTIVE NEURO-FUZZY INFERENCE SYSTEMS AND OPTIMIZATION ALGORITHMS

This chapter will provide an overview of the Neuro-fuzzy system, an introduction, and a definition of the theory of optimization, optimization algorithm, and Swarm Intelligence algorithms. Besides, this chapter covers the parameters and the steps of the selected SI-based algorithms. Finally, the tools used for the experiment will be discussed.

3.1. ADAPTIVE NEURO-FUZZY INFERENCE SYSTEMS (ANFIS)

An adaptive network-based fuzzy inference system (ANFIS) is a neuro-fuzzy system that blends artificial neural networks and fuzzy logic, has become gradually prominent in the field of Artificial Intelligence (AI). Neuro-fuzzy blends neural network learning and connectionist framework with a fuzzy system of human-like reasoning style. Fuzzy inference is the process of formulating input/output mappings using fuzzy logic. It has the potential to capture the benefits of both neural networks and fuzzy logic principles in a single framework. Its inference system is made up of a set of fuzzy IF-THEN rules with the ability to learn and approximate non-linear functions [28,29,30]. There are two areas in the field of neuro-fuzzy modeling research: Mamdani modeling, which is a linguistic fuzzy model based on interpretability, and Takagi-Sugeno-Kang (TSK), which focuses on accuracy. TSK is a non-linear static structure simulation instrument. ANFIS is based on the Takagi Sugeno fuzzy inference system. To interpret the input/output map, a network-type structure similar to a neural network may be used, which maps inputs through input membership functions and associated parameters to output membership functions, and then through output membership functions and associated parameters to outputs [29,30]. The membership functions (MFs) are the fundamental building elements of fuzzy set theory; the fuzziness of a fuzzy set is defined by its MFs. They can have a variety of forms, such as triangular, trapezoidal, Gaussian, bell shape, and so on [31]. For Fuzzy Inference Systems (FIS) generation, two states of subtractive clustering and grid partitioning were employed in estimating the parameters with ANFIS. Each data point is considered as a cluster center candidate in the subtractive clustering approach, and the potential of each data point is calculated by evaluating the density of data points surrounding it. In the grid partitioning method, the input data space is separated by an axis-parallel partition into a rectangular sub-space; each input is divided into identical membership functions. For membership function parameter estimation, the backpropagation variant of the steepest descent approach was used. The least-squares and backpropagation gradient descent methods are combined in the hybrid optimization technique [28,30,32].

The ANFIS architecture is given in Figure 2. It usually consists of a neural feed network of five layers [29,30].

Layer 1: This layer is the fuzzification layer. In order to generate fuzzy clusters from input data, this layer uses membership functions.

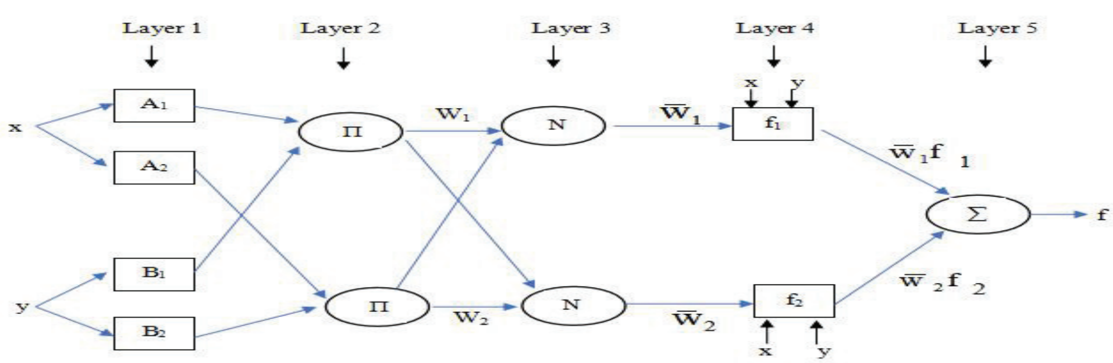


Figure 2: Basic structures of ANFIS [30]

Layer 2: This layer is referred to as the rule layer. Each node in this layer provides the strength of the rule using the multiplication operator. It executes the min (AND) operation.

Layer 3: This layer is referred to as the normalization layer. This layer normalizes the strength of all rules.

Layer 4: It is called adaptive or the defuzzification layer. The weighted values are calculated for the rules indicated in each node.

Layer 5: This is the output or the summation layer. Summing the outputs collected for each rule in the defuzzification layer yields the actual output of ANFIS.

3.2. OPTIMIZATION

Optimization is the process of estimating the most suitable solution to a problem by deciding the values of a set of parameters when the measure of optimality is achieved under given circumstances or subject to certain constraints. This activity has considerable significance to many professions, such as physicists, chemists, and engineers, who are interested in optimizing architecture [33]. The interest is to optimize the utilization of available resources, maximize organizational profit, or minimize efforts. Optimization applies both to minimization and maximization activities. There are three main components to an optimization problem given in [34] such as:

- Objective function: a function of the optimization problem that is supposed to be optimized to maximize or minimize. The task of maximization of the function *f* is equivalent to the study of minimizing *-f*, so the terms minimization, maximization, and optimization are used interchangeably.
- Variables: An objective function has, in terms of variables, several inputs.
- Constraints: put some restrictions or limitations on the variables.

3.3. OPTIMIZATION ALGORITHMS

Algorithms for optimization are used to find a set of independent variable values that optimize the value of one or more dependent variables. With arbitrarily high dimensionality, there are a variety of trivial multivariable optimization problems that cannot be solved in stimulating time by exact search methods [34].

One of the fields of Artificial Intelligence (AI) is Swarm Intelligence (SI), which is a new optimization technique that built on techniques that take advantage of the collective intelligence's features towards the solution of some tough problems that would have been difficult or impossible for a single agent to solve [28, 34, 35]. SI systems deal with the design of intelligent multi-agent systems by taking inspiration from the collective actions of social insects such as ants, termites, bees, and wasps and other animal communities, such as bird flocks, fish schools, and so on [35,36]. Such intelligence is decentralized, self-organized, and distributed throughout the environment [34]. Self-organization and division of labor are two key ideas that are regarded as necessary features of SI. Self-organization is defined as a system's capacity without any external support to create its agents or components into a suitable type. The second property is a division of labor, which is characterized as the simultaneous execution by individuals of various basic and feasible tasks. This separation helps the swarm to solve dynamic and complex problems requiring individuals to work together [37]. The key benefits of SI algorithms over traditional optimization strategies are: SI algorithms start from a set of possible solutions not from a single stage, do not need a derivative objective function, they are population-based, use multi-agents in their search processes, and agents that represent the population are homogenous [36,38]. The swarm intelligence algorithm features are practical and seek to improve the algorithm's deficiencies and improve the algorithm's performance. Due to their simplicity, versatility, and high performance they are also used to solve global optimization problems as primary techniques. The characteristics of swarm intelligence algorithm are also described in [38] as follows:

Strong robustness: The crowd control of the swarm intelligence algorithm is distributed; there is no central control. Thus, their working environment is in a wide variety; one or any individual issues should not affect the group.

Simple: It is quick and straightforward to run the execution of each independent operation.

Better scalability: Each sense has a limited amount of information

Strong self- organization: Individual interactions result in the group's complicated behavior.

3.4. SI OPTIMIZATION ALGORITHMS

The intelligent and cooperative behavior of social living organisms' triggers SI algorithms. Social creatures use this skill to search for food, security, prey, mating, and manage complicated scenarios [34]. Based on these strategies, there are different SI algorithms that are available. However, in this research based on the initial problem domain, exploration mechanism, and their applications, three algorithms that have been used to solve a variety of optimization problems such as Artificial bee colony (ABC), Genetic algorithm (GA) as well as Particle Swarm Optimization (PSO) are selected. Hence, this section provides an analysis and particular focus on their definition, parameters, and steps.

3.4.1. Artificial Bee Colony (ABC)

This algorithm, developed by D. Karaboga in 2005, is one of the recent popular swarm intelligence-based algorithms which is inspired by honey bee remarkable and intelligent behavior while looking for a supply of high-quality food, called nectar, and for the dissemination of information about that food source among other bees in the colony [34,37].

3.4.1.1. Elements of ABC Algorithm

The employed bees, unemployed bees, and food sources are three major elements of the ABC algorithm. To finish the algorithm's procedure, each of these bees has various tasks assigned to them.

1. The employed bees concentrate on a food source and remember the location of that food source.
2. The unemployed bees are onlookers and scout bees. The onlooker bee learns about the food source from the employed bee in the hive. The scout bee is responsible for locating new food sources and new nectar.
3. The third element is the excellent food sources nearby the beehive. Each food source is viewed as a potential solution to the problem of optimization [34,37].

The algorithm's search process has three main stages [34]:

- ❖ The employed bees are sent to the source of food and estimate their nectar quality.
- ❖ Onlooker bees choose the sources of food and calculate their quality of nectar approximately based on the information collected from employed bees.
- ❖ An employed bee whose food source has been abandoned becomes a scout and begins randomly searching for a new food source. The classification is governed by a control parameter known as limit or abandonment criteria.

3.4.1.2. Phases of ABC

Besides the above search processes, there are five steps of the ABC algorithm. These steps are initialization, employed bee, onlooker bee, scout bee phase, and the termination criteria as given below [34,37,39].

1. Initialization: The ABC algorithm has three major parameters: the number of food sources known as population, the number of tries after which a food source is treated to be dropped, which is known as the limit, and the termination criterion. The scout bees and control parameters are used to initialize all vectors of the food source population, X_i *(i = 1...SN, where SN is the population size).* Each food source, X_i, is a

solution vector consisting of n variables and it is a potential solution to the optimization problem under consideration. X_i, is given as follows [34,37,39]:

$$X_i = X_{i,min} + rand\left[0,1\right](X_{i,max} - X_{i,min}) \text{ ……………………………………………………… (3.1)}$$

Where: $X_{i,\,min}$, and $X_{i,\,max}$ are lower and upper bound of X_i, respectively.

: rand [0,1] is a function that generates an evenly scattered random number in the range of [0,1].

2. Employed Bee Phase: The search for a new food source, V_i, increases to have more nectar around the neighborhood of the food source, X_i. V_i is given as follows [34,37,39].

$$V_i = X_i + \phi_i\left(X_i - X_j\right) \text{ ……………………………………………………………… (3.2)}$$

Where: ϕ_i is a random number in the range of [-1,1].

: X_j is a randomly selected food source.

Once the new food source is produced, its fitness value is computed. If its fitness value is better than X_i the new food source replaces the previous one. The fitness value of the solution, $fit_i(X_i)$ is determined using the following equation [34,37,39]:

$$fit_i(X_i) = \begin{cases} \dfrac{1}{1+f_i(X_i)} & if \quad f_i(X_i) \geq 0 \\ 1+abs(f_i(X_i)) & if \quad f_i(X_i) \leq 0 \end{cases} \text{ ……………………………… (3.3)}$$

Where: $f_i(X_i)$ is the objective value of the solution (X_i).

3. Onlooker Bee Phase: determine the probability of selection for each food source provided by the employed bee. The probability P_p is calculated using each food source's fitness values in the population as follows [34,37,39]:

$$P_p = \frac{fit(X_i)}{\sum_{i=1}^{SN} fit(X_i)} \text{ ……………………………………………………………… (3.4)}$$

4. Scout Bee Phase: After a predetermined number of experiments, employed bees whose solutions may not be enhanced become scouts, and their solutions are plentiful. These scouts are starting to search for new solutions.

5. Steps 2-4 are repeated until termination criteria are satisfied.

The above steps are illustrated in the following figure.

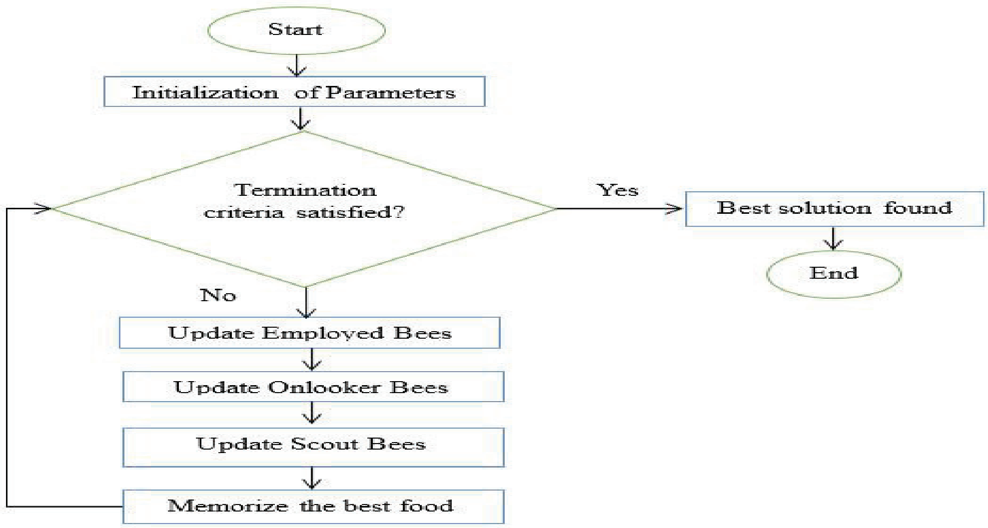


Figure 3: Flowchart of ABC [39]

3.4.2. Genetic algorithm (GA)

In the 1975s, John Holland first developed a Genetic algorithm (GA). It is an optimization search algorithm focused on the natural selection mechanism. The fundamental theory of GA is to mimic the concept of the survival of the fitness; i.e. the processes in the natural system are simulated, in which the strong tend to adapt and survive while the weak tend to vanish [37].

3.4.2.1. Components of GA

The main parts of GA, such as initialization, fitness assignment, genetic operators (selection, crossover, mutation, elitism), and termination are discussed below [40,41].

1. Initialization: When optimizing, a solution with a random initial value is usually started. A chromosome is an individual solution in a population, referring to the parameters to be optimized. The migration to a new population from the existing population is called generation. The generations reflect the population's iteration.

2. Fitness assignment: Fitness assignment means that a fitness value is assigned to each chromosome related to the objective function. These fitness values control the direction of search within the GA.
3. Genetic operators: In GA, a genetic operator is used to preserve genetic variation and combine current solutions. The basic genetic operators in GA are encoding, selection, crossover, and mutation.
    3.1 Encoding: The information given must be converted in a certain bit or string. According to the model of the problem, different types of encoding (binary encoding, octal, permutation encoding, hexadecimal, value encoding, and tree encoding) can be applied.
    3.2 Selection: This scheme determines whether or not the particular string is involved in the process of reproduction. The rate of GA convergence is determined by the selection pressure. Boltzmann, Rank, Roulette wheel, Stochastic universal sampling, Tournament, Random, and steady-state are among various methods of selection.
    3.3 Crossover: It is a process used for the recombination of the two individuals' genetic materials. Two strings are selected from the mating pool on a random basis for almost all crossover operators, and certain parts of strings between the strings to form two new chromosomes are swapped. One-point, two-point crossover, three-point crossover, multi-point crossover, and uniform crossover are among the crossover methods. One-point crossover is illustrated in the following figure.

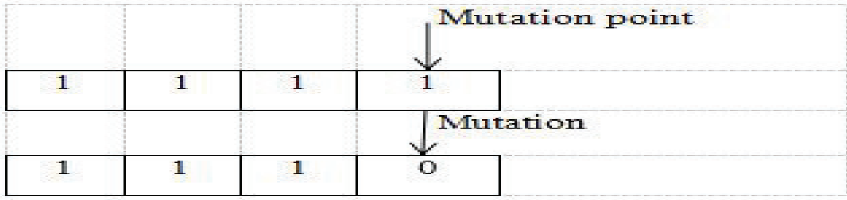


Figure 4: One-point crossover [40]

3.4 Mutation: It is a process by which genetic diversity is maintained from one population to the next. Bitwise mutation, Displacement, simple inversion, and scramble mutation are well-known mutation operators. The figure below depicts a bitwise mutation.

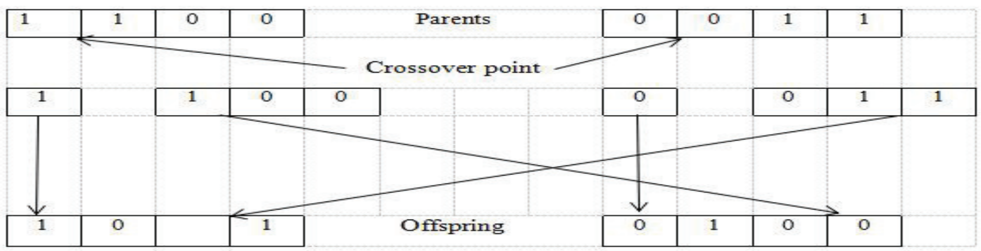


Figure 5: Bitwise mutation [40]

3.5 Elitism: It is the ability to store the best solutions found in the process.

4. Termination: It is normally carried out on two bases: the highest generation number is achieved, or the optimal fitness value for the best solution is achieved (specified by the user).

3.4.2.2. Steps involved in GA

In this algorithm, there are different steps as listed below [40,41]:

1. Randomly or heuristically describe an initial population.
2. For each member of the population, calculate the fitness value.
3. According to the fitness value, two chromosomes are selected from the population.
4. Based on the crossover probability crossover is applied to produce offspring.

5. Based on the mutation probability mutation is applied to the offspring to generate new offspring and placed in the new population.
6. Repeat these steps until the right solution is obtained.

The following figure depicts the steps and basic implementations of the GA.

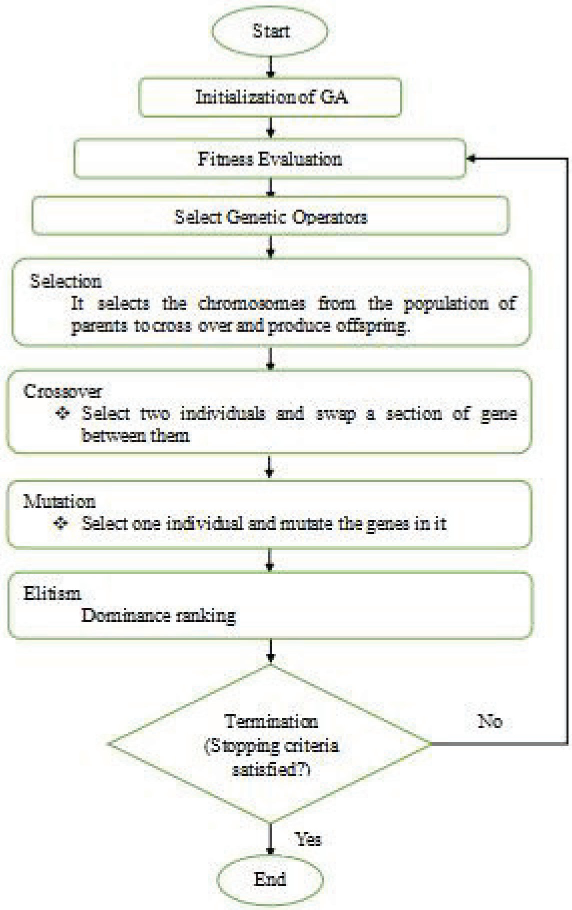


Figure 6: Basic implementation of GA

3.4.3. Particle Swarm Optimization (PSO)

This algorithm is discovered by Kennedy in 1995. The PSO uses the particles to find global optimal solutions via a simple approach that resembles swarm behavior in birds flocking and fish schooling sociological behavior. They seek the individual's best position and make sure all teams retain the optimum state according to their overall information by changing speed and position. Each particle is analogous to the bird and they regularly changing their position and speeds in the PSO process [38].

3.4.3.1. Parameters of PSO Algorithm

Swarm size or number of particles, the number of iterations, velocity components, and acceleration coefficients are the simple PSO parameters outlined below [33,38,39].

1. Swarm size: The number of particles in the swarm is swarm size or population size.
2. Iteration numbers: The number of iterations to achieve a successful outcome often depends on the problem.
3. Acceleration coefficients: In combination with random values r_1 and r_2, which are random numbers between 0 and 1, the acceleration coefficients C_1 and C_2 retain the stochastic effect of the cognitive and social parts of the particle velocity respectively. The constant C_1 represents how much confidence a particle has in itself, thus C_2 reflecting how much confidence a particle has in its neighbors.
4. Velocity Components: There are two extreme values in PSO used to determine position and velocity. The first extreme value is the optimal solution found by the particle itself, known as personal best (*pbest)* or local best (*lbest*), and the other extreme is the current global best *(gbest)*, which is the optimal solution found by the entire population. Three terms of the velocity of the particle arise in the following equations [33,38,39]:

$$V^{t+1}_{ij} = \omega\, V^{t}_{ij} + C_1 r_1 [p_{best,i} - X^{t}_{ij}] + C_2 r_2 [g_{best} - X^{t}_{ij}] \text{ ………..…...……..……...… } (3.5)$$

$$V^{t+1}_{ij} = \omega\, V^{t}_{ij} + C_1 r_1 [p_{best,i} - X^{t}_{ij}]\ + C_2 r_2 [p_{best,i} - X^{t}_{ij}] \text{ ....……………..…...…..…… } (3.6)$$

i. The term V^{t}_{ij} is called the part of inertia, which includes a memory of the previous direction of flight that indicates movement in the immediate past. The inertia weight, ω, is used to monitor the effect of the experience of velocities on the current speed, thereby, in turn, regulates particle exploration and exploitation behavior.

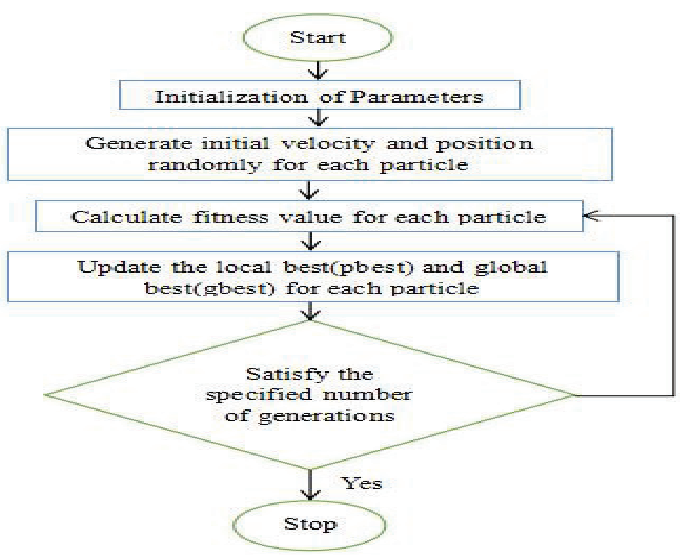
ii. The term $C_1 r_1\ [p_{best,i} - X^{t}_{ij}]$ is called the cognitive component, which tests the particles' performance compared to previous performances.

iii. The term $C_2r_2\,[g_{best}-X^t{}_{ij}]$ for *gbest* PSO or $C_2r_2\,[p_{best,i}-X^t{}_{ij}]$ for *lbest/pbest* is referred to as a social component that tests particle performance compared to a group of particles or neighbors.

3.4.3.2. Steps of PSO algorithm

The algorithm's steps are described below [33,38]:

1. Set the particles in the search space at some random velocities and positions;
2. Calculate the fitness value of the swarm particles accordingly;
3. Updating the individual and global best.
4. Next, the fitness value is equated with the previous best overall. If the value of the current is better than *gbest*, then reset *gbest* to the array index and value;
5. Assign these values to the swarm particle's respective position and velocity;
6. Finally, repeat steps 2-5 until stopping criteria are met.



The following figure shows the steps involved in the PSO algorithm.

Figure 7: Basic implementation of PSO [33].

3.5. TOOLS USED FOR THE EXPERIMENTS

For the comparison of optimization Algorithms, different types of the open-source software were used in this study. The Windows 10 Enterprise N Operating System (OS), Graphical Network Simulator-3 (GNS3), VMware workstation pro, and traffic generator tool (Ostinato), and MATLAB were addressed in the next subsection.

3.5.1. Windows 10 Enterprise N

Windows 10 Enterprise N Version 20H2 is an OS version designed to meet the requirements of large and medium-sized companies (including large academic institutions), such as advanced security protection against modern security threats, complete OS deployment flexibility, updating, and support options, as well as comprehensive management and monitoring capabilities for devices and applications [42]. In this study, the Windows 10 Enterprise N OS was running on the Lenovo Intel Core computer to manage the computer's memory and processes.

3.5.2. Graphical Network Simulator-3 (GNS3)

GNS3 (version 2.2.5) is an open-source and free software, a network software emulator first released in 2008, which runs the OS that emulates network hardware's real behavior. It has allowed network engineers to virtualize real hardware devices and support many devices from multiple vendors, and more devices are being added all the time [43]. To simulate, configure and test the network, GNS3 was employed in this research.

3.5.3. VMware workstation pro

VMware Workstation Pro (version 15.5.5) is the industry standard for running multiple operating systems as Virtual Machines (VMs) on a single Linux or Windows personal computer [43]. In this study, this software is integrated with the GNS3 to set up and runs the VMs on the computer.

3.5.4. Ostinato (0.9)

Ostinato is an open-source, cross-platform network packet traffic generator and analyzer with a friendly interface and network automation capabilities via a robust Python. It supports different protocols and the architecture of the client-server. It can generate and configure sequential and interleaved protocol streams at different rates. A user-defined script also allows the flexibility to incorporate any unimplemented protocol [44]. In this thesis, to generate the packets Ostinato was used.

3.5.5. MATLAB

MATLAB is a high-level programming language and interactive environment that allows us to accomplish computationally complex tasks more quickly than with traditional programming languages like C, C++, and Fortran. It efficiently expresses matrix and array of mathematics in order to conduct iterative analyses and design. It contains the live editor for creating scripts that integrate code, output, and formatted text in an executable notebook [45]. In this research, for the generation of the model, implementation, and comparison of optimization algorithms MATLAB was used.

4

EXPERIMENTAL ANALYSIS

This chapter explains how the dataset for the experimental study of the BGP slow convergence problem was generated, collected, and applied to the models. The first section describes the system model. The second section discusses the experimental setup. The third section provides a simulation of the existing network topology scenario. Finally, in the fourth section, we'll go through the model experimentation such as simulation of BGP scenario using GNS3 simulator, data generation method, generation of the model using the data-driven model as well as application of three optimization algorithms.

4.1 SYSTEM MODEL FOR REDUCTION OF BGP CONVERGENCE TIME

This section describes the system model for the reduction of BGP convergence time. In order to meet the goals of the study and answer the research questions, the following steps were followed.

In the first step: Create a network scenario based on the Ethio telecom backbone network and then the simulation is performed using GNS3.

In the second step: For the created scenario, BGP was configured so that the MRAI timer values were updated. The values start with 0 seconds for each MRAI and then increase by 1-second intervals until 600 was reached.

In the third step: The node failure event closer to the destination of the traffic with 600 different MRAI values produced 600 convergence time and 600 update messages results.

In the fourth step: This data was gathered to train the neuro-fuzzy system to provide the model.

In the fifth step: The optimization algorithms are applied to the model from the training system to solve the optimization problem then, evaluate the algorithms based on different metrics such as convergence rate and computational complexity such as Time elapsed and Memory usage.

In the sixth step: Based on the results, obtained from step five, the optimum value of the MRAI timer is selected from the best algorithm. Finally, BGP with the default MRAI value which is five seconds, and optimized MRAI value will be compared in terms of convergence time and packet loss.

The figure below depicts the framework of the thesis work from data collection to model experimentation and evaluation.

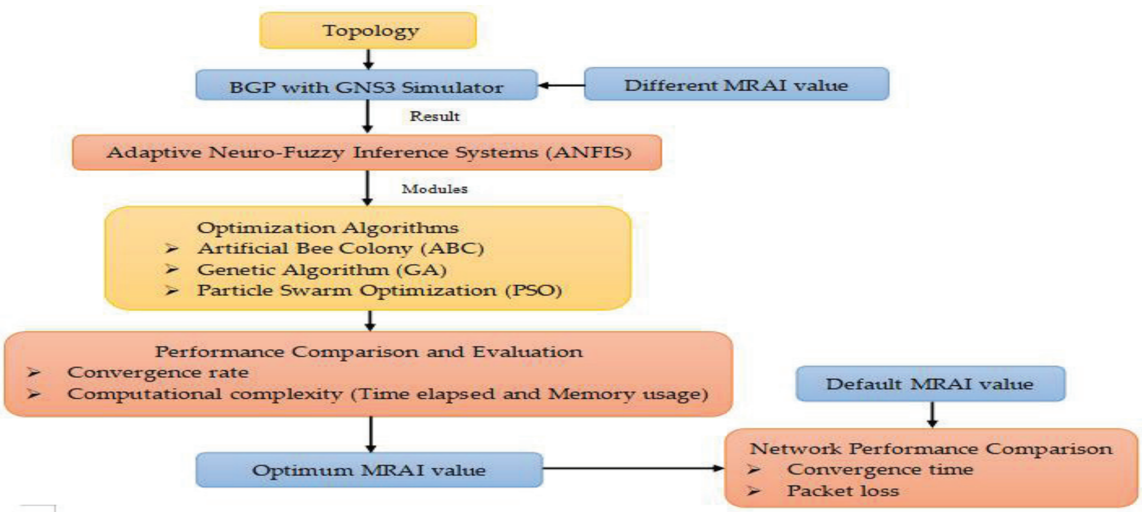


Figure 8: System Model for optimum MRAI value

4.2 EXPERIMENTAL SETUP

All experiments in this research were carried out on a computer with Lenovo Intel Core i7-8550U Central Processing Unit (CPU) running @ 3.00 GHz processor, 8 GB Random-Access Memory (RAM), and Windows 10, 64-bit operating system (OS). Graphical Network Simulator version 2.2.5 was employed to configure the network topology and for the data generation process. For the data modeling system and optimization algorithms, Matlab tools were used in the experiments.

4.3 SIMULATION SCENARIO AND NETWORK TOPOLOGY

In the implementation part, a practical environment is developed using a network simulation platform and a scenario is built to test the performance of BGP convergence time using default and optimized MRAI. The simulation tool used to simulate the network topology is the GNS3 version 2.2.5 open-source tool. The main reason to choose this tool is for its user-friendly Graphical User Interface (GUI) and the ability to configure and simulate a network component in a virtual machine that runs the same OS as the original network component. For network packet generation an open-source network packet/traffic generator and analyzer, ostinato, is employed. It was selected because it has a user-friendly graphical user interface and its capability to transmit packets from multiple streams using various protocols at varying speeds. As shown in Figure 9, the network topology consists of 12 routers, 2 switches, 1 packet generator, and 2 personal computers. Each component memory slot is with 256MB RAM. The Bole (BL) and Microwave (MW) sites are considered in the topology. The router BL-BR-A and BL-BR-B are the Border Routers (BR), the MW-ER-A, MW-ER-B, BL-ER-A, and BL-ER-B routers are serving as the Provider Edge (PE) or simply Edge Routers (ER). The rest of the routers such as MW-CR-A, MW-CR-B, BL-CR-RR-1 as well as BL-CR-RR-2 are the Core Routers (CR) and the Route Reflectors (RR) with redundant links to border routers respectively. The BL-CR-RR-1 and BL-CR-RR-2 are serving as both core router and RR. CE-1 and CE-2 routers are used as Customer Equipment (CE). To guarantee the service this network adopts double-plane architecture which is Plane A and Plane B. Service traffic will switch from Plane A to Plane B when the failure occurs. Access nodes and ER, are connected to local CR equipment directly and each pair of CR equipment is uplinked to two pairs of BR equipment of two sites in a ring.

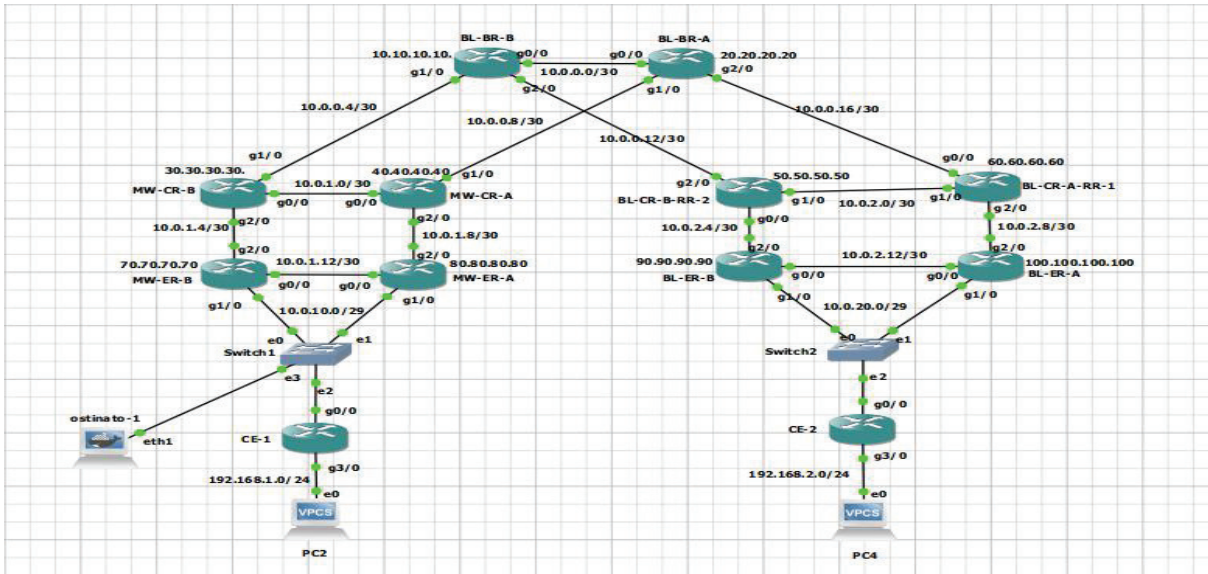


Figure 9: Network topology

4.3.1 Implementation of Interior Gateway Protocol (IGP) Design

The Intermediate System-to-Intermediate System (IS-IS) is a link-state routing protocol widely used in large Service Provider networks for its fast convergence, scalability, and efficient use of network bandwidth. The existing network topology IGP applies IS-IS level-2 protocol. IS-IS only carries routes of the interconnected port of BR, CR, ER, and RR, the detailed planning is as follows:

1. Enable IS-IS on all interfaces (including loopback interfaces) on all BR.
2. Enable IS-IS on the uplink interface of ER router (interfaces connected to CR routers).
3. Enable IS-IS on the interconnected interface between ER routers and the loopback interface.

The Virtual Router Redundancy Protocol (VRRP) is configured on Edge Routers. The VRRP function is to provide redundancy and to prevent a single point of failure inherent in the static default routed environment as well as to serve as a virtual default gateway for the CEs.

4.3.2 Implementation of BGP Design

As a service bearer network, it only requires all devices deployed in the network to keep detailed routes of overall network services, so that iBGP is run between all PEs, BR, and RRs. Each PE and BR establish iBGP neighborship with two RRs. Two RRs, redundancy as mutually backups, are responsible for route convergence, forwarding as well as reflecting VPN version 4 (VPNv4) route. The BGP is configured on RRs are given as shown in Figure 10.

```
BL-CR-A-RR-1#show running-config | begin bgp
router bgp 100
 bgp router-id 60.60.60.60
 bgp log-neighbor-changes
 neighbor 10.10.10.10 remote-as 100
 neighbor 10.10.10.10 update-source Loopback0
 neighbor 20.20.20.20 remote-as 100
 neighbor 20.20.20.20 update-source Loopback0
 neighbor 50.50.50.50 remote-as 100
 neighbor 50.50.50.50 update-source Loopback0
 neighbor 70.70.70.70 remote-as 100
 neighbor 70.70.70.70 update-source Loopback0
 neighbor 80.80.80.80 remote-as 100
 neighbor 80.80.80.80 update-source Loopback0
 neighbor 90.90.90.90 remote-as 100
 neighbor 90.90.90.90 update-source Loopback0
 neighbor 100.100.100.100 remote-as 100
 neighbor 100.100.100.100 update-source Loopback0
 !
 address-family ipv4
  no synchronization
  neighbor 10.10.10.10 activate
  neighbor 10.10.10.10 route-reflector-client
  neighbor 10.10.10.10 next-hop-self
  neighbor 20.20.20.20 activate
  neighbor 20.20.20.20 route-reflector-client
  neighbor 20.20.20.20 next-hop-self
  neighbor 50.50.50.50 activate
  neighbor 50.50.50.50 next-hop-self
  neighbor 70.70.70.70 activate
  neighbor 70.70.70.70 next-hop-self
  neighbor 80.80.80.80 activate
  neighbor 80.80.80.80 next-hop-self
  neighbor 90.90.90.90 activate
  neighbor 90.90.90.90 next-hop-self
  neighbor 100.100.100.100 activate
  neighbor 100.100.100.100 next-hop-self
  no auto-summary
 exit-address-family
 !
 address-family vpnv4
  bgp dampening
  neighbor 70.70.70.70 activate
  neighbor 70.70.70.70 send-community both
  neighbor 70.70.70.70 route-reflector-client
  neighbor 80.80.80.80 activate
  neighbor 80.80.80.80 send-community both
  neighbor 80.80.80.80 route-reflector-client
  neighbor 90.90.90.90 activate
  neighbor 90.90.90.90 send-community both
  neighbor 90.90.90.90 route-reflector-client
  neighbor 100.100.100.100 activate
  neighbor 100.100.100.100 send-community both
  neighbor 100.100.100.100 route-reflector-client
```

Figure 10: BGP configuration on RR

4.3.1 Multi-Protocol Label Switching-Virtual Private Networks (MPLS-VPN) Design

MPLS-VPN is a technology that enables a service provider to completely utilize parameters that are essential to offer their clients bandwidth, latency, and availability for service assurances. The technology provides for the establishment of Virtual Private Networks (VPNs) as well as scalability, to provide guaranteed service to its customers without the need for significant investments. The MPLS is configured globally on all PEs, RRs as well as on the IS-IS configured interfaces.

In a single router, VRF is a technology that supports many routing instances. It also permits the use of overlapping IP addresses because of its independence. It is commonly used combined with (MPLS-VPNs). The PE configures a VRF for each VPN to store the VPN user's routing table. Static route using CE as next-hop and the default route from CE to ER using VRRP as next-hop were configured on ER. And also all interfaces used for service access on ER are VRF bundling interfaces.

4.3.2 Generating traffic using ostinato

For this experiment, the Ethernet as Layer 2, IPv4 as Layer 3, and TCP as Layer 4 protocols are chosen. The selected column in figure 11 below indicates that traffic is being produced and sent at the desired rate. The source and destination are set as follows.

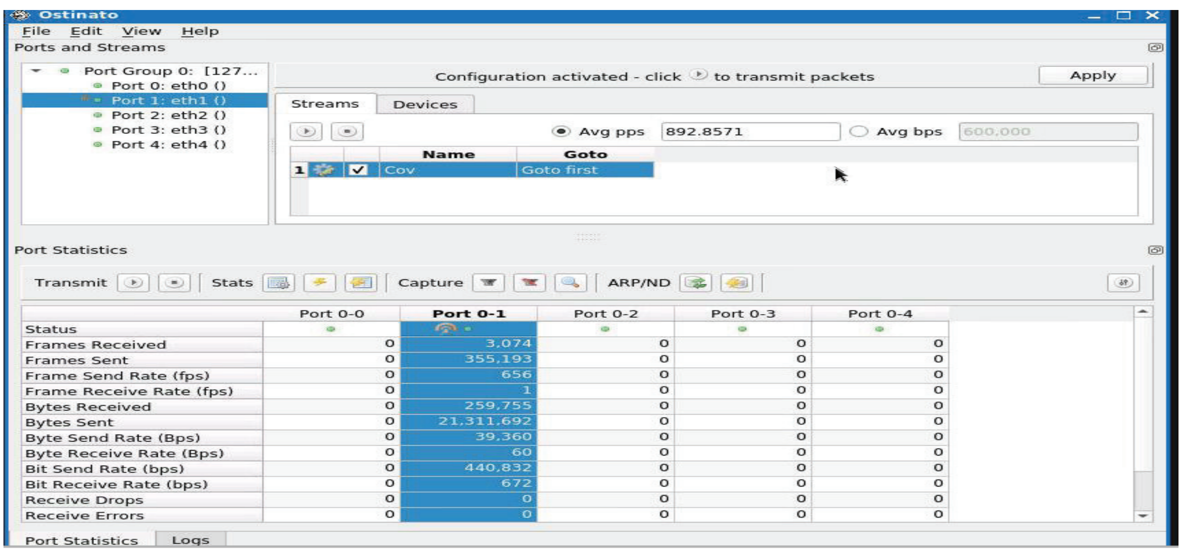


Figure 11: Traffic generation on Ostinato

4.3.3 DATA Generation

In data-driven models, a dataset is often needed to train the ANFIS model to build up mapping functions. Accordingly, in this study a network traffic dataset is collected from measurements of convergence time and the number of update messages under node failure closer to the destination of the traffic. The MRAI timer values are updated for the given scenario. For each MRAI, the values begin at zero seconds and increase in one-second intervals until 600 is reached. With 600 different MRAI values, node breakdown closer to the traffic's destination resulted in 600 convergence time and 600 update messages. The sample configuration is illustrated in the figure below.

```
BL-CR-A-RR-1
BL-CR-A-RR-1(config)#router bgp 100
BL-CR-A-RR-1(config-router)#address-family ipv4
BL-CR-A-RR-1(config-router-af)#neighbor 10.10.10.10 advertisement-interval 1
BL-CR-A-RR-1(config-router-af)#neighbor 50.50.50.50 advertisement-interval 1
BL-CR-A-RR-1(config-router-af)#exit
BL-CR-A-RR-1(config-router)#address-family vpnv4
BL-CR-A-RR-1(config-router-af)#neighbor 70.70.70.70 advertisement-interval 1
BL-CR-A-RR-1(config-router-af)#neighbor 80.80.80.80 advertisement-interval 1
BL-CR-A-RR-1(config-router-af)#neighbor 90.90.90.90 advertisement-interval 1
BL-CR-A-RR-1(config-router-af)#$0.100.100.100 advertisement-interval 1
BL-CR-A-RR-1(config-router-af)#
```

Figure 12: MRAI configuration on RR

The following is the experimental scheme for the data collection method:

1. The main path is CE-1->MW-ER-A-->MW-CR-A-->BL-BR-A-->BL-CR-A-RR-1-->BL-ER-A-->CE-2 as shown in Figure below.

```
CE-1#
CE-1#ping 192.168.2.1 repeat 50

Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to 192.168.2.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (50/50), round-trip min/avg/max = 324/548/992 ms
CE-1#
CE-1#trace 192.168.2.1

Type escape sequence to abort.
Tracing the route to 192.168.2.1

  1 10.0.10.3 120 msec 172 msec 140 msec
  2 10.0.1.10 [MPLS: Labels 34/38 Exp 0] 932 msec 808 msec 936 msec
  3 10.0.0.10 [MPLS: Labels 33/38 Exp 0] 940 msec 868 msec 704 msec
  4 10.0.0.18 [MPLS: Labels 37/38 Exp 0] 656 msec 992 msec 856 msec
  5 10.0.20.3 [MPLS: Label 38 Exp 0] 672 msec 636 msec 704 msec
  6 10.0.20.4 828 msec 732 msec 512 msec
CE-1#
```

Figure 13: The main path

2. A single node has been disconnected by the failure event for the simulated experiment and waits until the ping connectivity via an alternate route is restored. The ping statistics are calculated and recorded. The BGP router sent the update message since the adding or removal of a node triggered a routing change in the network. So that the number of update messages is recorded for each MRAI value. After the failure event, the communication is formed through the alternate route such as CE-1-->MW-ER-B-->MW-CR-B-->BL-BR-B-->BL-CR-A-RR-2-->BL-ER-B-->CE-2 as seen in figure 14.

```
CE-1#
CE-1#ping 192.168.2.1 repeat 50

Type escape sequence to abort.
Sending 50, 100-byte ICMP Echos to 192.168.2.1, timeout is 2 seconds:
!!!!!!!!!..........!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 80 percent (40/50), round-trip min/avg/max = 240/555/1012 ms
CE-1#
CE-1#trace 192.168.2.1

Type escape sequence to abort.
Tracing the route to 192.168.2.1

  1 10.0.10.3 36 msec 68 msec 76 msec
  2 10.0.1.13 [MPLS: Labels 32/37 Exp 0] 516 msec 484 msec 484 msec
  3 10.0.1.6 [MPLS: Labels 34/37 Exp 0] 520 msec 428 msec 464 msec
  4 10.0.0.6 [MPLS: Labels 22/37 Exp 0] 480 msec 624 msec 524 msec
  5 10.0.0.14 [MPLS: Labels 25/37 Exp 0] 448 msec 420 msec 588 msec
  6 10.0.20.2 [MPLS: Label 37 Exp 0] 452 msec 424 msec 424 msec
  7 10.0.20.4 464 msec 360 msec 520 msec
CE-1#
```

Figure 14: The Alternative path

3. Then the broken node is re-established, and connectivity is resumed over the main route.
4. The steps from 1 to 4 are repeated 600 times and calculate the convergence time in seconds for each case to get the data source.

4.4 MODEL EXPERIMENTATION

4.4.1 Adaptive Neuro-Fuzzy Inference System (ANFIS)

After data is generated and collected, the dataset has been ready for model training purposes. This implementation was done with the MATLAB tool. The key goal of this section is to learn and decide on the nature of BGP convergence time and the number of update messages. The intelligent system modeling tool used to accomplish this purpose is an ANFIS. The inputs to ANFIS are the average degree, the average diameter, number of nodes, and minimum route advertisement interval. The output is the convergence time in seconds. The data was transformed to fuzzy values to build the FIS component of the ANFIS system. The fuzzification process involves converting control input data from the system into symbolic values, which are linguistic qualifiers. By applying the membership function, the fuzzy set and membership degree to which the input information belongs to is identified, and the numerical value entered is assigned the linguistic variable values of low, medium, and high. To ensure the system's efficient functioning, fuzzy sets of various MFs forms can be used [29,30]. They are crucial to the overall performance of fuzzy representation. According to a comprehensive review of different literatures when multiple MFs are used to solve problems, triangular MFs are widely used, simple, performing well, and better than other MFs. Besides, three triangular MFs for each input and a zero-order output MF provide the best results. The Grid partitioning and hybrid learning algorithms were adopted for the ANFIS model training based on their high efficiency and accuracy [31,32]. Based on this, the membership function used in this experiment was a triangular MF for each input variable, a zero-order polynomial, as well as the default value for the remaining methods, are chosen for Sugeno-style fuzzy inference. All details are given in Table 1.

Table 1: FIS model Parameters

No	Parameters	Types
1	Fuzzy rule generation method	Grid Partitioning
2	Learning algorithm	Hybrid
3	FIS Type	Sugeno
4	Input MF	Triangle
5	Output MF	Zero-order polynomial
6	Number of inputs	4
7	Number of output	1

As shown in Figure 15 below, each of the three membership functions is consistent with the low level, medium level, and high level of the input levels respectively. The triangular membership function is used for each input. The output membership function used in this experiment was a zero-order polynomial for Sugeno-style fuzzy inference.

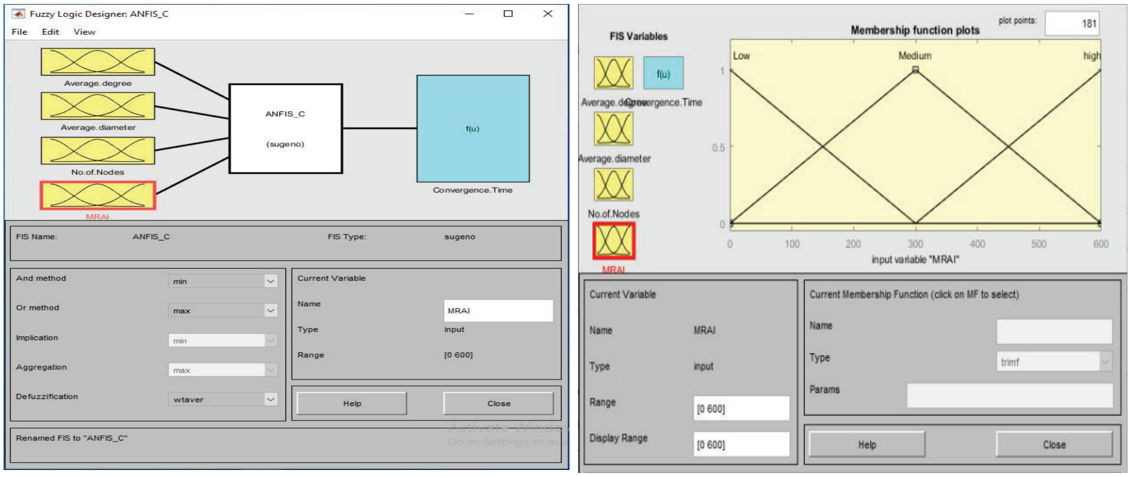


Figure 15: Sugeno FIS model

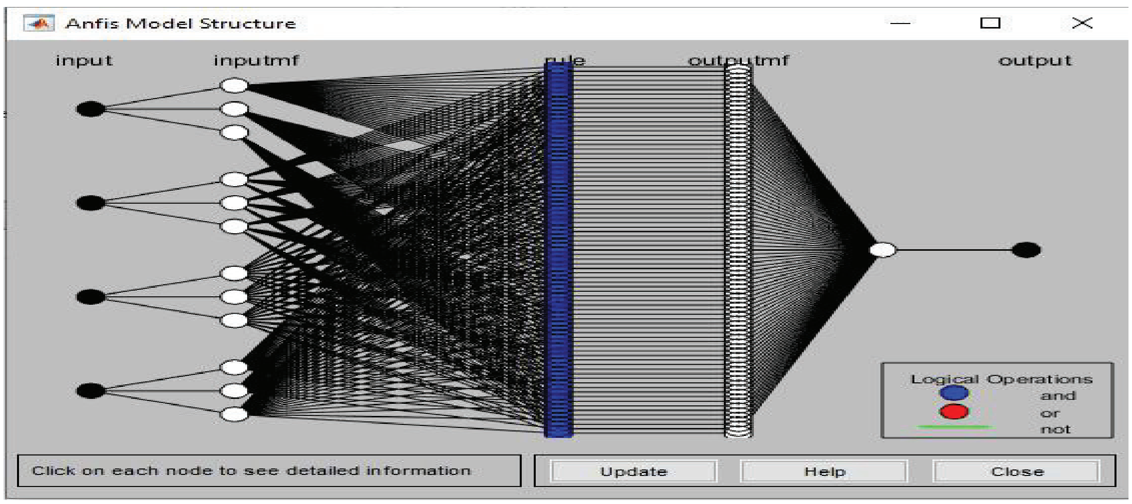


Figure 16: ANFIS structure

As shown in Figure 16 above, after the training is done the generated ANFIS structure is obtained. The structure contains 4 inputs in the first layer of the proposed model and a hidden layer in the second layer with three neurons corresponding to the input membership function. The blue color indicates the logical operator (AND) is the aggregation operator used, the number of nodes in each layer of the neural network and finally only one neuron for output or inference can be seen. The benefit of using this system is that an expert would not need the rules used by the inference system to be set. The necessary rules are generated from an input-output data set. Two models are produced as a result of this step: one for convergence time and the other for the number of update messages.

4.4.2 Optimization algorithms

The main objective of this thesis work is to evaluate the performance of optimization algorithms to solve the BGP slow convergence problem. Once the output of the ANFIS is acquired, optimization algorithms were applied to the model from the training system. The algorithms are used to optimize the parameters of the model, which means the MRAI were tuned using three optimization algorithms to minimize the convergence time without affecting the number of update messages. The implementation has been performed using the MATLAB tool.

4.4.2.1 A model with Artificial Bee Colony Experimentation

The first experiment has been implemented model with Artificial Bee Colony optimizer. As mentioned in Chapter 3, ABC is inspired by the remarkable behavior of honey bee insects while looking for a supply of high-quality food. The optimal fitness value for the best solution is achieved between all the founded values in all iterations. The ABC algorithm has three control parameters: 1) the colony size, which consists of employed bees and onlooker bees and 2) the limit value, which is the number of trials required for a food-source location to be abandoned. The study in [46] indicated that a colony size of (50-100) and a moderate value of limit for scouts such as 50% of the colony size provide a reasonable convergence speed and better performance. On this basis, the experimentation of this model has been done by employing the following parameters.

Table 2: Parameters of ABC used in the test scenario

No.	Parameters	value
1	Maximum Iteration	100
2	Colony size	50
3	Abandonment Limit Parameter	25

4.4.2.2 A model with Genetic Algorithm Experimentation

The second experiment has been implemented a model with a Genetic Algorithm. As describes in chapter 3, it maintains a pool of chromosomes, which can develop with the selection, crossover, and mutation operations. The optimal fitness value for the best solution is achieved between all the founded values in all iterations. Three fundamental parameters are used by GA, including 1) Crossover probability which determines the number of times a crossover occurs in the chromosomes in one generation; 2) Mutation probability this rate determines how many genes mutate in one generation, and 3) the size of the population refers the total number of the population's inhabitants. The study in [40,47] suggested that optimal population size ranges of (50–100), mutation parameter rate of (0.01-0.15), and the crossover employed was based on one-point crossover at a rate of (0.8). Based on this, the following parameters have been used in the experimentation of this model.

Table 3: Parameters of GA used in a test scenario

No.	Parameters	value
1	Maximum Iteration	100
2	Population size	50
3	Crossover probability	0.8
4	Mutation probability	0.15

4.4.2.3 A model with Particle Swarm Optimization Experimentation

The third experiment has been implemented a model with a PSO classifier. As mentioned in Chapter 3, PSO is based on bird-flocking sociological behavior. The birds adjust the location and direction of the flight continuously to find an individual's best position. The global value for the best solution is achieved between all the founded values in all the iterations. The particle swarm optimization algorithm is determined by four parameters such as 1) Inertia weight; 2) Acceleration coefficients for cognitive; 3)

Acceleration coefficients for social and 4) Number of populations. According to [33], the values of the number of population, C_1, C_2, and ω as 50, 1.7, 1.7, and 0.7, respectively are recommended for better results. Based on this, the following parameters have been used in the experimentation of this model.

Table 4: Parameters of PSO used in the test scenario

No.	Parameters	value
1	Maximum Iteration	100
2	Number of Population	50
3	Inertia weight	0.7
4	Acceleration coefficients for cognitive	1.7
5	Acceleration coefficients for social	1.7

4.5 OPTIMIZATION ALGORITHM EVALUATION METRICS

When handling various optimization issues due to their specific characteristics, optimization algorithms perform differently. Several special sessions and competitions are held to assess the performance of algorithms in dealing with complicated challenges. The following parameters are used to evaluate the performance of the algorithms [48,49]:

1. Convergence rate/speed: in the context of optimization algorithms the iterative sequence generated by the algorithm converges to the optimal method. It is one of the fundamental factors for the performance evaluation of the algorithm.
2. Computational complexity: An algorithm's computational complexity is a measure of the amount of computing resources that it consumes as it executes. The complexity of the algorithm is divided into two such as:

   2.1. Time complexity/running time: The number of operations executed determines how long an algorithm takes to run for a certain input. An algorithm's computational time increases as the number of operations increases. The running time aids in the comprehension of evolutionary algorithms, the evaluation of efficiency, and the improvement of algorithms.

2.2 Space/ Memory complexity: The amount of storage space or memory required by an algorithm to execute as a component of the length of the input is measured by its space complexity.

4.6 NETWORK PERFORMANCE EVALUATION METRICS

The performance of a network refers to the measure of a network's service quality as experienced by the user. Depending on the nature and design of the network, there are several techniques to assess network performance. The parameters that are used to assess a network's performance include:

1. Convergence time: the period between the initial service interruption and the return of full data flow is known as convergence time. This is a critical metric to assessing network availability and resilience for service providers [7,13,22]. The formula for calculating convergence time is shown in Equation (4.1).

$$Convergence\,Time = Average\ round\ trip\,time \times (\,failover\,time - faildown\ time\,) \text{ .... } (4.1)$$

2. Packet loss: The datagram drops that occur in the route of a one-way traffic flow between the source and destination node are referred to as packet loss. Datagrams may be lost as a result of network element failures until the failure is detected and the connection is restored. Equation (4.2) shows the formula for determining the percentage of packet loss [50]:

$$Packet\ loss = \frac{Packet\ sent - Packet\ received}{Packet\,sent} \times 100\% \text{ .... } (4.2)$$

5

RESULTS AND DISCUSSION

This chapter presented all the results obtained from testing and evaluating the selected three optimization algorithms to solve BGP slow convergence problem. Furthermore, this chapter compares the models using different performance metrics. The results of the tests were discussed based on this experimental analysis.

5.1 EXPERIMENTAL RESULTS

5.1.1 Simulation Results

In this section, the results such as the BGP convergence time and the number of update messages from the network simulation platform were collected to train the Neuro-fuzzy system.

The convergence time is the period between the initial service interruption and the restoration of full data flow. When all impacted routes have shifted from the main to the alternative way, network route convergence is supposed to be complete. During simulation for calculating the BGP convergence time, the Internet Control Message Protocol (ICMP) ping test is used on the given network. The ping test is used to determine the connection between hosts on the network, the total amount of time the source takes to send a packet to the destination, and the time it takes to send the response back to the source. Figures 17 & 18 show the sample simulation output of convergence time and the number of update messages respectively. From this output, convergence time is computed as the average Round Trip Time (RTT) multiplied by the time from the outage until the connection has been established. Besides, the number of update messages was also collected for each case. For instance, for 2 seconds MRAI value, the convergence time is calculated as (0.646sec*20sec) =12.92 sec.

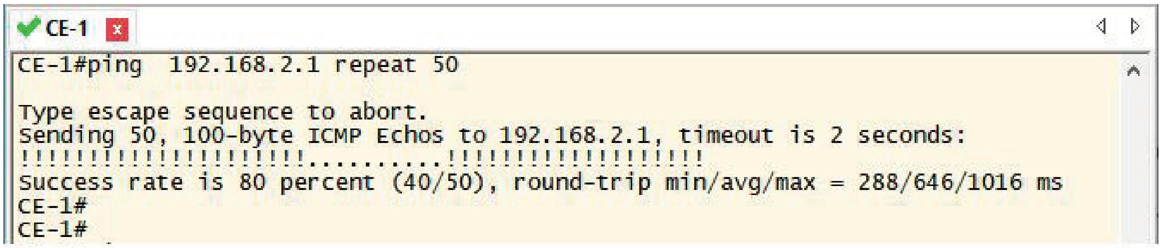


Figure 17: Convergence time

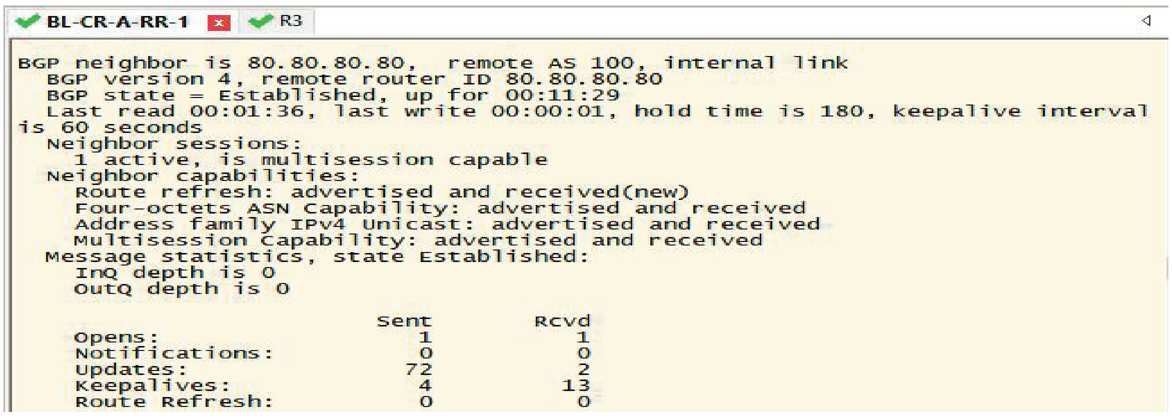


Figure 18: Number of update messages on RR

For simulation single node failure event is considered. The sample test results for different MRAI values are tabulated for the given scenarios as shown in Table 5. Then for the ANFIS model, a total of 600 sample tests were used.

Table 5: Sample convergence time for different MRAI values

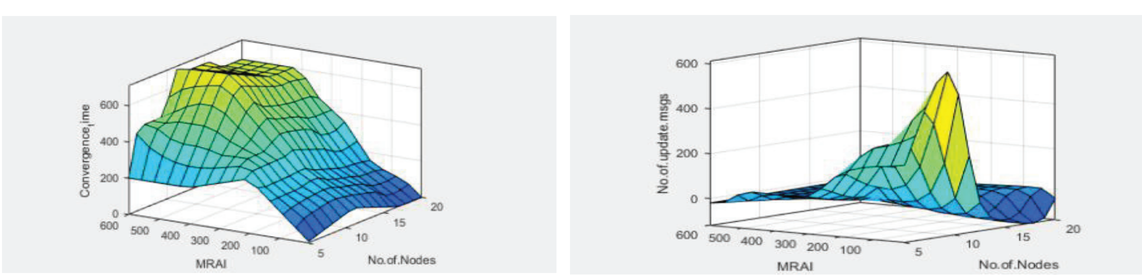
MRAI values(sec)	0	1	2	3	4	5	6	7	8	9	10
Convergence time (sec)	3.7	6.1	12.92	19.2	25.7	28.8	36.2	43.2	48.7	53.8	59.2
Number of update messages	92	80	72	72	72	72	72	72	70	70	70

5.1.2 ANFIS's Result

This section presents the implementation and the output of the ANFIS. All in all, around 600 MRAI values of convergence time and the number of update message data were captured. It was able to train this system to generate an accurate model using this type of data. The ANFIS had four inputs; each one had three membership functions which have low, medium, and high level. During the training process, 3^4=81 rules can be automatically generated to represent the input-output relationships of the data as shown in the Figure below.

Figure 19: ANFIS generated rules for convergence time

As a result of the implementation of this step, the model for the convergence time and the number of update messages is given below.



(a) (b)

Figure 20: Model of First-Order ANFIS for (a) Convergence time, (b) Number of update messages

Figure 20 depicts the dependence surface, which is a portable function of the chosen model. It is clear that the MRAI timer in the network has a greater effect on reducing BGP convergence delays. In terms of the effect of this timer on the observed output, it is concluded that the convergence time in the network increases with the increase of the specified variable and number of nodes. An increase in the number of update messages is also caused by a reduction in the MRAI timer and an increasing number of nodes.

5.1.3 Optimization Results

In this section, the results of the optimization for three applications are presented. The algorithms were applied to the convergence time and number of update messages modules from the implementation result of ANFIS. To evaluate the performance of each model's parameters such as the rate of convergence against the number of execution to achieve its known best value; the average running time which is necessary to finish the algorithm and the average memory usage required to run the operations of the algorithm are selected. These evaluation parameters are the most significant factors for optimization algorithms to be recognized as the best models for the BGP slow convergence problem. Each approach was subjected to fifty independent trial runs with 100 iterations.

Figure 21 shows the graphical representation of the convergence rate of the optimization model's performance. According to this, the PSO algorithm has a consistent solution and faster convergence speed towards the global optimum solution followed by GA. Whereas ABC has the lowest rate of convergence as compared to GA and PSO.

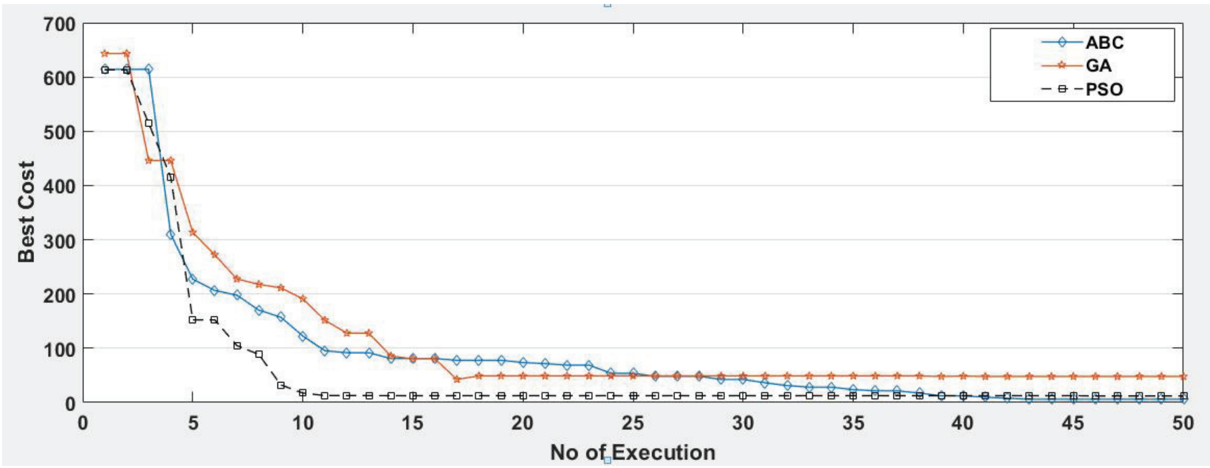


Figure 21: Comparison of convergence rate

The time taken to complete ABC, GA, and PSO for various iterations in a generalized normalized environment with the same processor and memory was compared. It is necessary to note that the machine abilities and the implementation of the optimization techniques affect the execution time. On the other hand, memory usage refers to how much memory an algorithm consumes in order to finish the searching process. The use of a high memory implies that the system memory is being wasted. Table 6 presents the evaluation results of ABC, GA, and PSO algorithms in terms of average time and average space consumption for 50 independent executions with Iteration 100. Accordingly, the computational time and the space taken by the ABC algorithm to search for the global optimum are 8.95 sec and 778KB respectively. Whereas GA consumes 5.84 sec running time and 2636KB memory usage. The computational time and space required by the PSO algorithm to perform its operation are 3.96 sec and 803KB respectively.

Table 6: Performance Comparison of Selected Algorithms

Selected Algorithm	Memory usage (KB)	Running time (sec)
ABC	778	8.95
GA	2636	5.84
PSO	803	3.96

As shown in the above result, the PSO algorithm took minimum execution time, whereas, in the ABC optimization model, the maximum time is taken to reach an optimal solution. The GA models presented a comparatively medium value of running time. The following figure illustrates the time complexity of each model at the same condition.

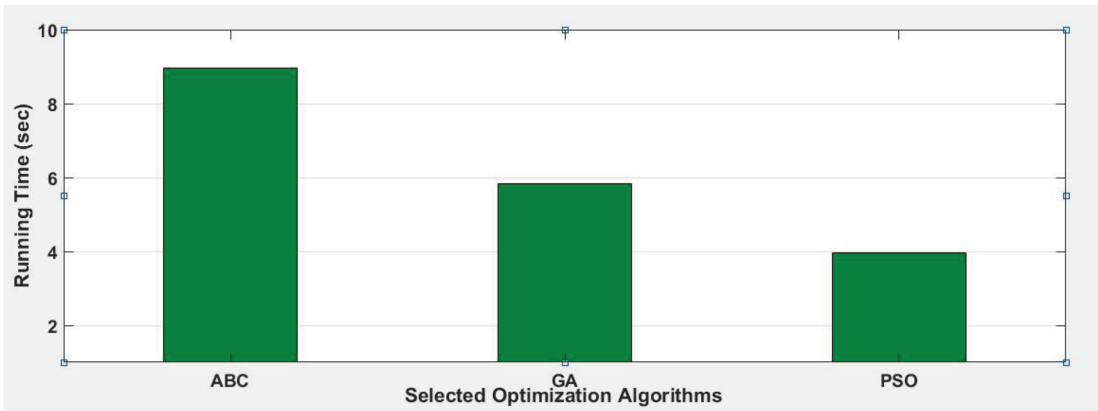


Figure 22: Comparison of time complexity

On the other hand, the space that GA occupied is much higher than ABC and PSO. ABC and PSO use the lowest memory usage than GA but PSO have a comparably large amount of memory as compared to ABC. Figure 23 reflects the graphical representation of memory usage by each model at the same task.

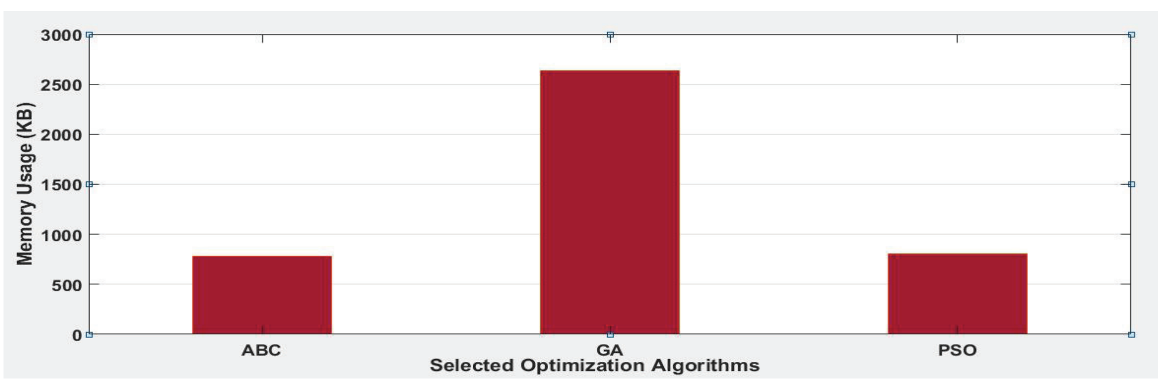


Figure 23: Comparison of space complexity

In summary, the optimization models building experiment was accomplished in the previous chapter. Experiments were done using the MATLAB tool.

ABC shows a significant advantage in terms of space complexity for the average number of executions required. However, in terms of convergence rate and running time, due to numerous operations that occur during its iteration, ABC performs poorly. GA has the lowest computational time as compared to ABC but it has premature convergence and uses a high level of memory usage. PSO has a faster rate of convergence because of its comprehensive and extensive search capabilities, but it took a greater amount of memory as compared to ABC.

Finally, by observing the simulation studies and results, except the memory consumption, PSO achieves the best results in all evaluation metrics than the others two techniques. It is successful in obtaining a near-optimal solution in a short amount of time and at a short convergence rate. As a result, it can be concluded that the model which is developed with the PSO optimization approach is considered as the selected working model for BGP slow convergence problem. Accordingly, the optimal MRAI is selected from the PSO algorithm. The PSO determined that two seconds was the best value for the MRAI because the convergence time was reduced with regardless of affecting the number of update messages. Based on this, Figure 24 depicts the result for the network with 17 nodes. One of the nodes failed after 2 minutes, and the BGP protocol was required to converge. The default BGP with the default MRAI value (five seconds) took 29 seconds before it converges. However, the optimized BGP with the recommended MRAI value (two seconds) reached 13 seconds, indicating that the optimized BGP converged faster than the default, the delay was minimized.

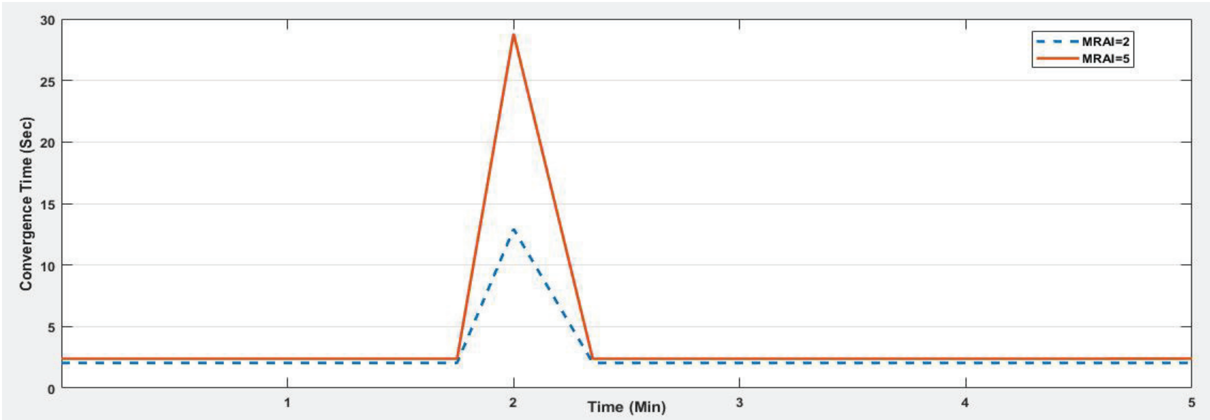


Figure 24: Comparison between convergence time with default and optimum MRAI

Table 7: Packet loss for default and optimized BGP

Data Size (bytes)	36	1000	2000	3000	4000	5000	6000
Packet Loss (%) for default BGP	0	2	8	20	36	46	55
Packet Loss (%) for optimized BGP	0	2	4	6	8	17	36

As shown in Table 7 there is no packet loss in both scenarios when the nodes are not disconnected but, in both cases, the packet drop increases very fast when the node is being failed. The performance of optimized BGP is much better than that of default BGP. The graphical representation is given in Figure 25.

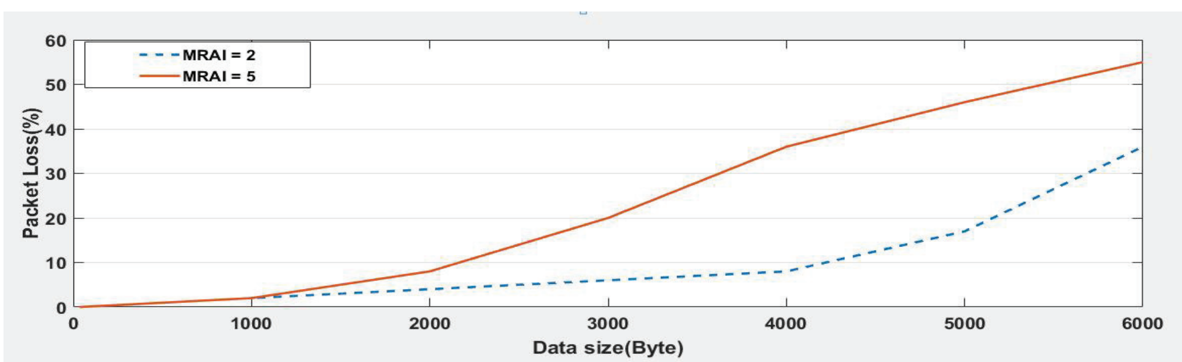


Figure 25: Comparison of packet loss for default and optimized BGP

5.2 DISCUSSION

In this thesis work the major aim was to evaluate the performance of optimization algorithms as described in chapter 1 in order to solve BGP slow convergence problem and minimizing convergence time as well as packet loss. To achieve this objective and answer the research question dataset is generated and collected from the simulated network and the input-output mapping has been done using ANFIS. Furthermore, from different literature reviews, recommendations, initial problem domain, exploration mechanism, and their applications the SI optimization algorithms such as ABC, GA, and PSO have been selected and implemented on the MATLAB tool.

The first experiment was conducted on BGP simulation with different MRAI timers (from zero to six hundred seconds), for each simulated test, a single node failure event has been carried out. The BGP router needed to converge and transmitted the update message as this resulted in a network routing change. As a result, the convergence time and the number of update messages are collected and used as a dataset to train the ANFIS.

The second experiment has been performed with the ANFIS-TSK model to develop a mapping function. The result in Figures 19 & 20 shown that the model has been mapped the input-output relationship by generating rules and the resulted models provided that with an increase in the MRAI timer and number of nodes, network convergence time increases. A decrease in the MRAI timer and an increasing number of nodes is also causing an increase in the number of the update message.

The third experiment has been carried out with the ABC optimization algorithm. The result in Figure 21 shown that the rate of convergence for all models under the same scenario. The convergence of the optimal solution in the ABC method is quite slow. All models have comparable performance metrics, as indicated in Table 6. It achieves the lowest memory requirement than the others two algorithms but it took a longer execution time due to its computational complexity depends on the dimension of the problem, the number of iterations, and the number of populations.

The fourth experiment with the GA optimization algorithm was conducted. As shown in Table 6 and Figure 21, all of the models have similar performance metrics. It took less running time to search for optimal value than ABC but it suffers from premature convergence and a high level of memory requirement since its computational complexity is determined by the genetic operators, how they are implemented, how individuals and populations are represented, and the fitness function.

The fifth experiment has been performed with a PSO optimization algorithm. The convergence rate for all models under the same circumstance was illustrated in Figure 21 and in Table 6; all models have comparable performance metrics. PSO can converge to the optimum value with reduced computational time and achieves a faster rate of convergence when compared to the other two optimization algorithms since its computational complexity depends on the population and the problem dimension. However, it has larger and smaller memory requirements than ABC and GA respectively.

As per the results obtained from the above experiments, the PSO optimization algorithm has better performance and works very well as compared to the aforementioned algorithms. It converges at iteration of 8, took approximately 4 seconds and consumed 803KB of memory usage. The algorithm returned that two seconds as the optimal MRAI value since it decreased the convergence time maximally while without change in the number of update messages.

The last experiment was conducted using BGP simulation with default and optimal MRAI timer. For each simulated test, a single node failure event has been carried out to compare the performance of a network regarded to convergence time and packet loss. Single network element failure is used as a factor in this convergence time and packet loss analysis, which means that the node in the network is deliberately disconnected from the network. This allows us to compare the time taken until the BGP system is stable in both network conditions. During this time each router will re-calculate alternative routes and build a new routing table. After convergence is accomplished the packet transmission from source to destination resumes. For the packet loss analysis ping test were sent using the ICMP echo with different data sizes. For instance, at a data size of 6000 bytes, there is 19% additional packet loss in default BGP. Optimized BGP has significantly greater performance than default BGP as it shortens the time required for convergence this will, in turn, minimize the packet loss.

This thesis work addresses the following research questions and achieves the goals outlined in Chapter 1. As previously stated, the primary goal of this research was to assess the performance of three optimization algorithms for the BGP convergence delay problem.

1. Which optimization technique can be more effective and achieve better performance to solve BGP slow convergence problem?

   Speed of convergence and complexity of the selected optimization strategies are compared. Convergence speed and complexity are defined by having on an average fewer number of iterations, a shorter running time, and low memory usage to achieve an optimal solution respectively. And also, the robustness is measured by the capacity to achieve a consistent solution in multiple independent runs in the search processes. Based on this, due to its robustness, fast convergence and low time complexity PSO exhibits the best performance and has been chosen as the best algorithm for BGP slow convergence problem.

2. Which MRAI value is optimum for better network performance?

   As several researchers indicated that the reduced MRAI timer has a contribution to the reduction of routing convergence in order to avoid or minimize loss of packets during failures. The delay in convergence time also has a negative impact on the performance of a network. Accordingly, the PSO algorithm identified two seconds as an optimal value as it maximally decreased the convergence time and packet loss. Therefore, two seconds of the MRAI timer is proposed in this thesis.

6

CONCLUSION AND FUTURE WORK

According to the results presented in chapter 5, this chapter derived a conclusion and provided recommendations for future work.

6.1 CONCLUSION

The main objective of this thesis was to compare optimization algorithms to identify optimum MRAI timers to improve BGP slow convergence time. To achieve this objective, the network topology was configured using GNS3 with the required configuration files. In the subsequent step, network traffic is generated using Ostinato. The simulation data were collected and used as a dataset to train the Neuro-fuzzy system which in turn provides the models. Next, three optimization algorithms were applied to the models using the MATLAB tool. To do the analysis and comparison of the applied algorithms three evaluation parameters such as rate of convergence, computational time, and space/memory usage were used to evaluate the algorithms. Accordingly, the optimum value for the MRAI timer was identified from the algorithm with better performance. Finally, the network performance of BGP with default and optimized MRAI value were compared using convergence time and packet loss.

For this research, the optimization algorithms such as ABC, GA, and PSO were utilized for analysis. The test results show that the PSO algorithm converges to optimal value with reduced computational time when compared to ABC and GA. However, in the case of space required to run the operation, PSO took less than GA and higher than the ABC algorithms. Except for its memory usage (as compared to ABC), the model using the PSO performs better in all evaluation performance measures. For the optimization of the MRAI timer, GA has moderate convergence rate and time usage than ABC but it consumes larger space than others.

The following conclusions can be drawn from the study and simulation results:

- The comparative results of the three algorithms show that PSO outperformed the other two algorithms in obtaining optimum value.
- The MRAI timer affects convergence time and packet loss. This indicates as it can be optimized to improve the convergence time and packet loss and found that two-second as an optimized MRAI timer.
- Compared to the default MRAI timer, the optimized MRAI improves the delay in convergence up to 55%.
- At the same node failure event optimized MRAI reduces packet loss by 19%.
- This study demonstrates that the BGP is highly sensitive to fine-tuning of its parameters.
- Any service provider, including Ethio telecom, can implement the optimum MRAI timer in BGP configuration to improve the quality of service for the company's customers and overall network performance by speeding up packet rerouting, reducing packet loss and delay in the event of failure.

6.2 RECOMMENDATION FOR FUTURE WORKS

The following areas are recommended for future work.

- More investigations on the combination and optimization of other BGP timers such as KEEPALIVE, HOLD timer with MRAI timer which enhances the overall network performance can be done.
- Future research may focus on the hybridization of the aforementioned algorithms to improve the search ability of PSO which detects the global optimum value of BGP parameters (mentioned in the above) for slow convergence problems in short running time and low memory usage.

REFERENCES

[1] R. Bennesby and E. Mota, “A survey on approaches to reduce inter-domain routing convergence delay in the Internet,” *IEEE Communications Surveys & Tutorials*, Vol 19, Issue 4, 2017.

[2] J. Nykvist, “Convergence Issues in Inter-Domain Routing and Connectivity Modeling of Ad hoc Networks,” *Luleå University of Technology, Department of Computer Science and Electrical Engineering,* 2006.

[3] J. Park, “Understanding the Impact of Internal BGP Route Reflection,” P*h.D. dissertation*, *University of California*, 2011.

[4] S. Shukla and M. Kumar, “Improving Convergence in iBGP Route Reflectors,” *Lecture Notes in Electrical Engineering Advances in Electronics, Communication, and Computing*, pp. 381–388, Springer, Singapore, 2017.

[5] M. Caesar, et.al., “Design and Implementation of a Routing Control Platform,” *2nd Symposium on Networked Systems Design & Implementation*, pp. 15–28, 2005.

[6] C. Labovitz, et.al., “Delayed Internet routing convergence,” *In IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 293-306, 2001.

[7] S. Deshpande and B. Sikdar, “On the impact of route processing and MRAI timers on BGP convergence times,” *IEEE Global Telecommunications Conference* (GLOBECOM) Vol.2, pp. 1147-1151, Dallas, TX, 2004.

[8] A. Fabrikant, et.al., “There's something about MRAI: Timing diversity exponentially worsens BGP convergence,” Proc. of INFOCOM, Shanghai, iBGP, China, 2011.

[9] E. Rekhter, et.al., “A Border Gateway Protocol 4 (BGP-4),” *RFC 4271.1970*. [Online]. Available: https://www.https://tools.ietf.org/html/rfc4271. [Accessed: 09-Dec-2020].

[10] H. Devikar, et.al., “Impact of MRAI Timer on BGP Updates and Convergence Time,” *Indonesian Journal of Electrical Engineering and Computer Science* 12, pp.873-

882, 2018.

[11] ETC IP NGN Project, and Z. Min, "ETC IP NGN Project Implementation Plan V3.0", 2008.

[12] R. Devikar, et.al., "Study of BGP Convergence Time," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, no. 1, pp. 413-420, India 2016.

[13] P. Singh, "Analysis of MRAI Timer on BGP Convergence," *International Journal of Scientific Progress and Research (IJSPR) ISSN*: 2349-4689, Issue 158, Vol 58, No 01, pp 24-27, Rewa, 2019.

[14] E. Alabdulkreem, et.al., "MRAI Optimization for BGP Convergence Time Reduction without Increasing the Number of Advertisement Messages," *Procedia Computer Science*, vol. 62, pp. 419–426, London, 2015.

[15] S. Shukla and M. Kumar, "Optimized MRAI Timers for Border Gateway Protocol in Large Networks," *International Journal of Distributed Systems and Technologies*, vol. 10, no. 4, pp. 31–44, 2019.

[16] N. Laskovic, "BGP with an adaptive minimal route advertisement interval," *Simon Fraser University*, Canada, 2007.

[17] R. Paul, "Performance Evaluation of Border Gateway Protocol with Route Flap Damping and Routing Policies," *Thesis, Library and Archives Canada Bibliothèque et Archives Canada,* Ottawa, 2014.

[18] C. White, "A Neural Network Approach to Border Gateway Protocol Peer Failure Detection and Prediction," *The Faculty of California Polytechnic State University-San Luis Obispo*, Dec. 2009.

[19] B. Ricardo, "DeepBGP: A Machine Learning Solution to reduce BGP Routing Convergence Time by Fine-Tuning MRAI," *Universidade Federal do Amazonas*, Doctor of Informatics, Brasil, 2019.

[20] H. Alshamrani, "Detecting IP prefix hijack events using BGP activity and AS connectivity analysis," *University of Plymouth Research Theses*, 2017.

[21] M. Vadakkepalisseri and K. Chandrashekaran, "Border Gateway Routing Protocol Convergence Time Analysis with Minimum Route Advertisement Information," *Communications in Computer and Information Science Information Processing and Management*, pp. 310–319, 2010.

[22] B. Wang, "The research of BGP convergence time," *2011 6th IEEE Joint International Information Technology and Artificial Intelligence Conference*, pp.354-357, 2011.

[23] E. Alabdulkreem, H. Al-Raweshidy, and M. Abbod, "Using a fight-or-flight mechanism to reduce BGP convergence time," *Fourth International Conference on Communications and Networking*, ComNet-2014, 2014.

[24] J. Qiu, F. Wang, and L. Gao, "BGP rerouting solutions for transient routing failures and loops," *In Military Communications Conference*, IEEE, pp. 1–7, 2006.

[25] J. John, E. Katz-Bassett, A. Krishnamurthy, T. Anderson, and A. Venkataramani, "Consensus routing: The internet as a distributed system," *In Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'08, (Berkeley, CA, USA), USENIX Association, pp. 351–364, 2008.

[26] S. Agarwal, C. Chuah, S. Bhattacharyya, and C. Diot, "The impact of BGP dynamics on intra-domain traffic," *In Proceedings of ACM SIGMETRICS*, vol. 32, (New York, NY, USA), ACM, June 2004.

[27] A. Wahid et.al., "Comparative Study of Routing Strategies in Software Defined Networking," *The 31st Annual ACM Symposium*, Pisa, Italy, pp.696-702, April 2016.

[28] A. Ahmed and S. Shah, "Application of adaptive neuro-fuzzy inference system (ANFIS) to estimate the biochemical oxygen demand (BOD) of Surma River," *Journal of King Saud University - Engineering Sciences*, vol. 29, no. 3, pp. 237–243, 2017.

[29] D. Karaboga and E. Kaya, "Adaptive network-based fuzzy inference system (ANFIS) training approach: a comprehensive survey," *Artificial Intelligence Review*, vol. 52, no. 4, pp. 2263–2293, 2018.

[30] M. Buragohain, “Adaptive Network-based Fuzzy Inference System(ANFIS) as a Tool for System Identification with Special Emphasis on Training Data Minimization,” *thesis, Indian Institute of Technology Guwahati*, Guwahati, 2009.

[31] A. Sadollah, “Introductory Chapter: Which Membership Functions Appropriate in Fuzzy System?” *IntechOpen,31-Oct-2018*. [Online]. Available: https://doi.org/10.5772/intechopen.79552. [Accessed: 23-Jun-2021].

[32] I. Ebtehaj and H. Bonakdari, “Performance Evaluation of Adaptive Neural Fuzzy Inference System for Sediment Transport in Sewers,” *Water Resources Management*, vol. 28, no. 13, pp. 4765–4779, 2014.

[33] F. Bergh, “An analysis of particle swarm optimizers,” *Thesis, University of Pretoria*, Pretoria, 2001.

[34] S. Kumar, “Nature-Inspired Computation Methods and Their Application in Function Optimization,” *thesis, Faculty of Engineering and Technology*, Jagannath University,2014.

[35] C. Blum, and X. Li, “Swarm Intelligence in optimization,” *Springer*, Berlin, Heidelberg, pp.43-85, January 2008.

[36] A. Chakraborty and A. Kar, “Swarm intelligence: A review of algorithms,” *Modeling and Optimization in Science and Technologies*, vol. 10, pp. 475–494, March 2017.

[37] A. Wahab, M. Nadhir, S. Nefti-Meziani, and A. Atyabi. “A comprehensive review of swarm optimization algorithms,” *PLoS one*, vol .10, no.5, pp. 1–36., Catalin Buiu, Politehnica University of Bucharest, ROMANIA, May 2015.

[38] Y. Ting, et al. “Swarm intelligence optimization algorithms and their application,” *Association for Information Systems AIS Electronic Library (AISeL)*, pp. 201-206, 2015.

[39] A. Hussein and R. Khalid, “A Comparative Study of Swarm Intelligence-Based Optimization Algorithms in WSN,” *Asian Journal of Engineering and Applied*

Technology, vol. 8, no.3, pp. 1-7, 2019.

[40] M. Rani, “Multi-objective optimization of PID controller parameters using genetic algorithm,” *Thesis, Universiti Teknologi Malaysia*, December 2012.

[41] S. Katoch, S. Chauhan, and V. Kumar, “A review on genetic algorithm: past, present, and future. Multimedia Tools and Applications”, *Computer Science and Engineering Department, National Institute of Technology*, Hamirpur, India, 2020.

[42] “Simon Fraser University Engaging the World,” *Windows 10 editions compared Computing Science- Simon Fraser University*. [Online]. Available: https://www.sfu.ca/computing/about/support/csil/windows/about-windows10 editions.html. [Accessed: 19-Feb-2021].

[43] “Getting Started with GNS3: GNS3 Documentation,” *GNS3 Docs*. [Online]. Available: https://docs.gns3.com/docs/#what-is-gns3. [Accessed: 19-Feb-2021].

[44] S. Mishra, S. Sonavane, and A. Gupta, “Study of traffic generation tools,” *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, vol. 4, no. 6, pp. 159–162, Jun. 2015.

[45] “MATLAB,” *MathWorks*. [Online]. Available: https://www.mathworks.com/products/matlab.html. [Accessed: 25-Mar-2021].

[46] D. Karaboga and B. Basturk, “On the performance of artificial bee colony (ABC) algorithm,” *Applied Soft Computing*, vol. 8, no. 1, pp. 687–697, 2008.

[47] A. Hassanat, K. Almohammadi, E. Alkafaween, E. Abunawas, A. Hammouri, and V. Prasath, “Choosing Mutation and Crossover Ratios for Genetic Algorithms—A Review with a New Dynamic Approach,” *Information*, vol. 10, no. 12, pp. 390, 2019.

[48] M. Asif, M. Khan, S. Abbas, and M. Saleem, “Analysis of Space & Time Complexity with PSO Based Synchronous MC-CDMA System,” *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2019.

[49] V. Beiranvand, W. Hare, and Y. Lucet, "Best practices for comparing optimization algorithms," *Optimization and Engineering*, vol. 18, no. 4, pp. 815–848, 2017.

[50] W. Sugeng, J. Istiyantp, K. Mustofa, and A. Ashari, "The Impact of QoS Changes towards Network Performance," *International Journal of Computer Networks and Communications Security*, vol. 3, no. 2, pp. 48-53, Feb. 2015.

APPENDIX

Performance Evaluation of Optimization Algorithms for BGP Slow Convergence Problem

Mihret Gebre[1*], Yalemzewd Negash[2]
[1,2] *School of Electrical and Computer Engineering, Addis Ababa Institute of Technology*
Addis Ababa, Ethiopia
[*]Mihretgebrenat2@gmail.com
yalemzewdn@yahoo.com

***Abstract*— Border Gateway Protocol (BGP) is an inter-domain routing protocol that provides routing or reachability information inside or between different Autonomous Systems (AS). The existing inter-domain routing architecture has a major challenge due to slow convergence during network failures. Convergence time can be characterized in terms of Minimum Route Advertisement Interval (MRAI) rounds. This paper is conducted to evaluate and analyze the performance of three optimization algorithms such as Artificial Bee Colony (ABC), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) for slow convergence problem that aims to find the optimum value of MRAI which reduces the BGP convergence delay, limit the effect of large-scale Internet failures as well as improve network performance. The results indicated that the PSO model outperforms the ABC and GA which reduces the running time and lowers the convergence rate required to complete the search process. The experimental results and analysis show that in optimized BGP convergence time is improved by 55%, and packet loss is improved by 19% as compared to default BGP.**

Keywords: *BGP, Convergence time, MRAI, ANFIS, Optimization Algorithm, Performance.*

I. INTRODUCTION

The Internet is composed of different separately regulated network domains known as Autonomous System (AS). The AS is a collection of routers under a single technical administration [1]. Every AS has a unique identifier called an Autonomous System Number (ASN). Border Gateway Protocol (BGP) executes routing on the AS network [2]. It is the de-facto standard inter-domain and a path-vector routing protocol on the Internet that links tens of thousands of networks on the Internet to create one huge network that's interconnected [1].

According to [3] during network outages, the existing inter-domain routing architecture faces a significant challenge which is slow convergence. This problem was addressed by different approaches such as using redundant standby paths to ensure the performance of the data plane during routing convergence, minimizing Minimum Route Advertisement Interval (MRAI) timer to accelerate routing propagation, and designing effective route flap damping to prevent update flooding with minimized MRAI time value. Convergence time is an important factor in defining routing protocol performance, and routing protocol is a key aspect in defining the quality of Internet Protocol (IP) communication. Convergence time is a period needed by the routing protocols to re-transmit the packet upon the failure or routing change. As a result, a network's convergence time is critical, and networks that converge quickly are seen as more stable [4,5].

The MRAI timer is used in internal BGP (iBGP) and external BGP (eBGP) sessions to limits the number of messages propagated by BGP speakers. The BGP speaker announces updates as the MRAI timer expires. Moreover, MRAI plays an essential role in BGP since the convergence time can be characterized in terms of MRAI rounds [5,6]. The default value of MRAI is 30 and 5 seconds for eBGP and iBGP respectively[7]. This paper examines the performance comparison of three optimization algorithms to overcome the BGP slow convergence problem to determine the optimum value for MRAI timer in iBGP Route-Reflector architecture.

The paper is structured into six sections and organized as follows. Section 2 focuses on the state-of-the art and literature review to provide a fundamental understanding of BGP, Section 3 introduces basic concepts in Adaptive Neuro-Fuzzy Inference Systems (ANFIS) and optimization algorithms that are used in this study. Section 4 deals with the system model. Section 5 focuses on experimental results and analysis of the outcomes of the experiment. Finally, Section 6 provides the conclusion of the study and suggests recommendations for further work.

II. BACKGROUND

This section presents the state-of-the-art and literature review for this research.

A. BGP States and Messages

To start communication between BGP peers, BGP defines the method of creating a BGP session. The BGP session can be in Idle, Connect, Active, Open Sent, Open Confirm, or Established state. The router does not set up a BGP session in the idle state and waits for the BGP start event. In the connected state, the router establishes the TCP connection between the BGP peers. In the active state, BGP waits for a TCP session from its neighbor. To complete a BGP session, the open sent and open confirmed states are used. On the other hand, in order to run BGP, BGP routers should form a neighbor relationship with adjacent BGP speakers. If two BGP peers are connected, four distinct types of messages may be exchanged to each other: OPEN, KEEPALIVE, UPDATE, and NOTIFICATION [2]. The OPEN message is the first message that the BGP speakers have shared. The UPDATE message can involve removal to a route that no longer exists or a new best path announcement. When any errors are occurred in a BGP message or during session establishment among BGP peers, the peer that identified the error, resulting in at the end of the session, sends a NOTIFICATION message. To ensure that the session persists, the KEEPALIVE messages are exchanged at a predetermined interval. In addition to these, BGP has a timer that limits event notification which is known as the MRAI. Before transmitting new updates, an update message should wait for some time equal to the MRAI value [8,9,10].

B. *BGP convergence delay*

In case of topological shifts due to link/node failure or policy changes, the routers might change their state from one correct and stable routing table condition to another correct and stable state by exchanging routing updates. This process is called convergence, and also the time taken by this process is named convergence time [11]. One reason why the Internet isn't so available is the convergence delay [1,12]. Different events can affect the BGP convergence process. There are two types of events in terms of topological changes, such as failure or recovery. Failure causes certain nodes/links to be inaccessible, and recovery returns some nodes/links to the system. Furthermore, failures can have two effects on the destination's reachability such as failover and fail down. In fail down, the destination from the network is disconnected by failure whereas in failover the nodes are forced to use an alternative path to reach the destination [11, 13].

Moreover, several factors influence BGP's convergence time, such as the MRAI timer, the number of update messages interchanged between BGP routers, adopted topologies, timers, and overhead of routers during recovery. On the other hand, different causes, such as TCP disconnection, security problems, and malfunction incidents, will make the interdomain service inaccessible [26]. Improving convergence can help to re-route the packets more efficiently, reduce packet loss and delay during failure [9, 15,16].

C. *Minimum Route Advertisement Interval Timer(MRAI)*

A BGP router divides the transmission of successive advertisements to the peer for a certain network; it uses MRAI to limit the number of update messages [17]. During convergence, the MRAI time attempts to speed up convergence toward bad behavior, which means transient routing, forwarding changes, and high overheads for routers. BGP speaker checks if the MRAI is not expired for that peer until it begins sending updates to any peer [9].

The effect of MRAI value on convergence delay will be illustrated below [6]:-

As the MRAI increases, nodes must wait longer before additional updates are sent, which will prolong the convergence delay. The number of updates sent in this process remains constant, and the time delay linearly increases.

If MRAI is smaller, updates are produced faster, but the router Central Processing Unit (CPU) load increases. So a node will give a neighbor an update before all update messages are processed. When one of the other updates changes the routes, it has just been advertised on the queue, and another update must be forwarded. The workload on nodes will be increased, which will, in turn, increase the delay in convergence. Thus, the optimum MRAI timer implementation plays a vital role in improving the convergence time of BGP.

III. ADAPTIVE NEURO-FUZZY INFERENCE SYSTEMS AND OPTIMIZATION ALGORITHMS

This chapter will provide an overview of the Neuro-fuzzy system, and Swarm Intelligence algorithms.

A. *Adaptive Neuro-Fuzzy Inference Systems (ANFIS)*

An adaptive network-based fuzzy inference system (ANFIS), is a neuro-fuzzy system that blends artificial neural networks and fuzzy logic, has become gradually prominent in the field of artificial intelligence (AI). It has the potential to capture the benefits of both neural networks and fuzzy logic principles in a single framework. Its inference system is made up of a set of fuzzy IF-THEN rules with the ability to learn and approximate non-linear functions [18,19,20]. There are two areas in the field of neuro-fuzzy modeling research: Mamdani modeling, which is a linguistic fuzzy model based on interpretability, and Takagi-Sugeno-Kang (TSK), which focuses on accuracy. TSK is a non-linear static structure simulation instrument. ANFIS is based on the Takagi Sugeno fuzzy inference system. To interpret the input/output map, a network-type structure similar to a neural network may be used, which maps inputs through input membership functions and associated parameters to output membership functions, and then through output membership functions and associated parameters to outputs [19,20].

B. *Swarm Intelligence (SI) Optimization Algorithms*

The intelligent and cooperative behavior of social living organisms' triggers SI algorithms. Social creatures use this skill to search for food, security, prey, mating, and manage complicated scenarios [21]. Based on these strategies, different SI algorithms that are available. However, in this research, based on the initial problem domain, exploration mechanism, and their applications, three algorithms that have been used to solve a variety of optimization problems such as Artificial bee colony (ABC), Genetic algorithm (GA) as well as Particle Swarm Optimization (PSO) are selected. Hence, this section provides an analysis and particular focus on their definition, parameters, and steps.

C. *Artificial Bee Colony (ABC)*

This algorithm, developed by D. Karaboga in 2005, is one of the recent popular swarms intelligence-based algorithms which is inspired by honey bee remarkable and intelligent behavior while looking for a supply of high-quality food, called nectar, and for the dissemination of information about that food source among other bees in the colony [21,22].

Steps of ABC

There are five steps of the ABC algorithm. These steps are initialization, employed bee phase, onlooker bee phase, scout bee phase, and the termination criteria as given below [21, 22, 23].

1. Initialization: The scout bees and control parameters are used to initialize all vectors of the food source population, X_i *(i = 1...SN, where SN is the population size).* Each food source, X_i, is a solution vector consisting of *n* variables and it is a potential solution to the optimization problem under consideration. X_i, is given as follows [21, 22, 23]:

$$X_i = X_{i,min} + rand\left[0,1\right](X_{i,max} - X_{i,min}) \quad (1)$$

Where: $X_{i,\ min}$, and $X_{i,\ max}$ are lower and upper bound on X_i, respectively.

: rand [0,1] is a function that generates an evenly scattered random number in the range of [0,1].

2. Employed Bee Phase: The search for a new food source, V_i increases to have more nectar around the neighborhood of the food source, X_i. V_i is given as follows [21, 22, 23].

$$V_i = X_i + \phi_i\left(X_i - X_j\right) \quad (2)$$

Where: ϕ_i is a random number in the range of [-1,1].

: X_j is a randomly selected food source.

Once the new food source is produced, its fitness value is computed. If its fitness value is better than X_i the new food source replaces the previous one. The fitness value of the solution, $fit_i(X_i)$ is determined using the following equation [21, 22, 23]:

$$fit_i(X_i) = \begin{cases} \dfrac{1}{1+f_i(X_i)} & if\ \ f_i(X_i) \geq 0 \\ 1+abs(f_i(X_i)) & if\ \ f_i(X_i) \leq 0 \end{cases} \quad (3)$$

Where: $f_i(X_i)$ is the objective value of the solution (X_i).

3. Onlooker Bee Phase: determine the probability of selection for each food source provided by the employed bee. The probability *Pp* is calculated using each food source's fitness values in the population as follows [21, 22, 23]:

$$P_p = \frac{fit(X_i)}{\sum_{i=1}^{SN} fit(X_i)} \quad (4)$$

4. Scout Bee Phase: After a predetermined number of experiments, employed bees whose solutions may not be enhanced become scouts, and their solutions are plentiful. These scouts are starting to search for new solutions.

5. Steps 2-4 are repeated until termination criteria are satisfied.

The above steps are illustrated in the following figure.

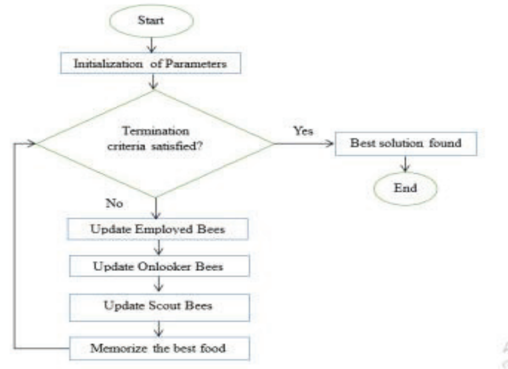


Fig. 1. Flowchart of ABC [23].

D. Genetic algorithm (GA)

In the 1975s, John Holland first developed a Genetic algorithm (GA). It is an optimization search algorithm focused on the natural selection mechanism. The fundamental theory of GA is to mimic the concept of the survival of the fitness; i.e the processes in the natural system are simulated, in which the strong tend to adapt and survive while the weak tend to vanish [22].

Steps involved in GA

In this algorithm, there are different steps as listed below [24, 25]:

1. Randomly or heuristically describe an initial population.
2. For each member of the population, calculate the fitness value.
3. According to the fitness value, two chromosomes are selected from the population.
4. Based on the crossover probability crossover is applied to produce offspring.
5. Based on the mutation probability mutation is applied to the offspring to generate new offspring and placed in the new population.
6. Repeat these steps until the right solution is obtained.

The following figure depicts the steps and basic implementations of the GA.

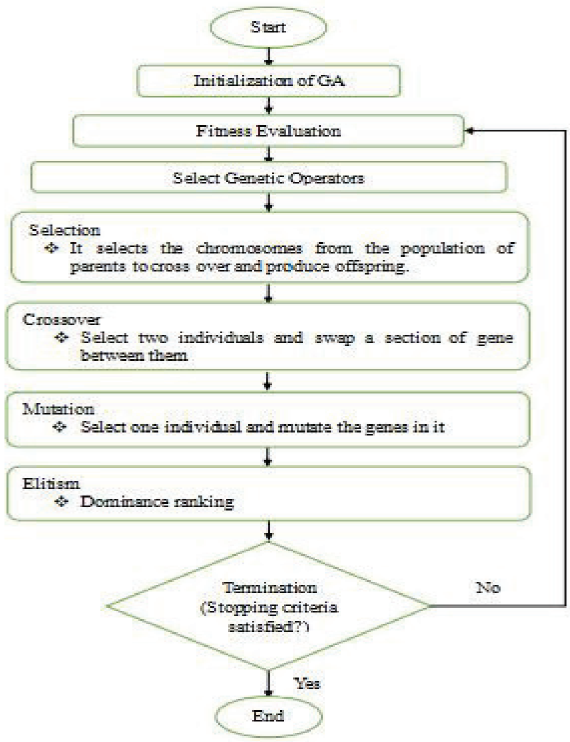


Fig. 2. Basic implementation of GA.

A. Particle Swarm Optimization (PSO)

This algorithm is discovered by Kennedy in 1995. The PSO uses the particles to find global optimal solutions via a simple approach that resembles swarm behavior in birds flocking and fish schooling sociological behavior. They seek the individual's best position and make sure all teams retain the optimum state according to their overall information by changing speed and position. Each particle is analogous to the bird and they regularly changing their position and speeds in the PSO process [26].

Parameters of PSO Algorithm

Swarm size or number of particles, the number of iterations, velocity components, and acceleration coefficients are the simple PSO parameters outlined below [23, 26, 27].

1. Swarm size: The number of particles in the swarm is swarm size or population size.
2. Iteration numbers: The number of iterations to achieve a successful outcome often depends on the problem.
3. Acceleration coefficients: In combination with random values r_1 and r_2, which are random numbers between 0 and 1, the acceleration coefficients C_1 and C_2 retain the stochastic effect of the cognitive and social parts of the particle velocity respectively. The constant C_1 represents how much confidence a particle has in itself, thus C_2 reflecting how much confidence a particle has in its neighbors.
4. Velocity Components: There are two extreme values in PSO used to determine position and velocity. The first extreme value is the optimal solution found by the particle itself, known as personal best (pbest) or local best (lbest), and the other extreme is the current global maximum (gbest), which is the optimal solution found by the entire population. Three terms of the velocity of the particle arise in the following equations [23, 26, 27]:

$$V_{ij}^{t+1} = \omega\, V_{ij}^{t} + C_1 r_1 [p_{best,i} - X_{ij}^{t}] + C_2 r_2 [g_{best} - X_{ij}^{t}] \quad (5)$$

$$V_{ij}^{t+1} = \omega\, V_{ij}^{t} + C_1 r_1 [p_{best,i} - X_{ij}^{t}] + C_2 r_2 [p_{best,i} - X_{ij}^{t}] \quad (6)$$

The term V^t_{ij} is called the part of inertia, which includes a memory of the previous direction of flight that indicates movement in the immediate past. The inertia weight, ω, is used to monitor the effect of the experience of velocities on the current speed, thereby, in turn, regulates particle exploration and exploitation behavior.

The term $C_1 r_1$ $[pbest_{,i}\text{-}X^t_{ij}]$ is called the cognitive component, which tests the particles' performance compared to previous performances.

The term $C_2 r_2$ $[gbest\text{-}X^t_{ij}]$ for gbest PSO or $C_2 r_2$ $[pbest_{,i}\text{-}X^t_{ij}]$ for lbest/pbest is referred to as a social component that tests particle performance compared to a group of particles or neighbors.

Steps of PSO Algorithm [26, 27]:

1. Set the particles in the search space at some random velocities and positions;
2. Calculate the fitness value of the swarm particles accordingly;
3. Updating the individual and global best.
4. Next, the fitness value is equated with the previous best overall. If the value of the current is better than *gbest*, then reset *gbest* to the array index and value;
5. Assign these values to the swarm particle's respective position and velocity;
6. Finally, repeat steps 2-5 until stopping criteria are met.

The following figure shows the steps involved in the PSO algorithm.

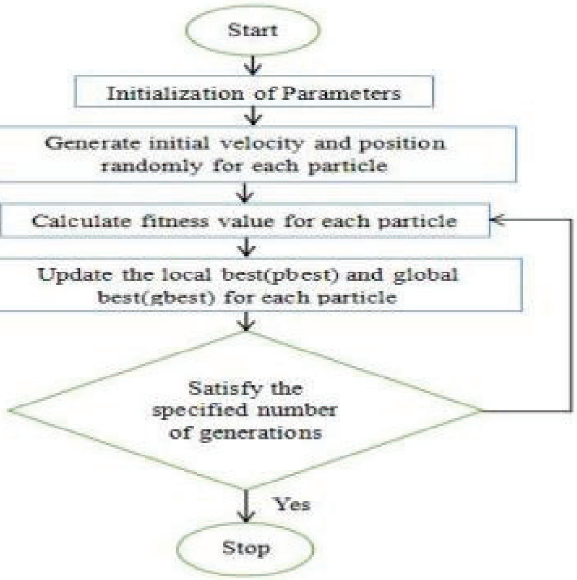


Fig. 3. Basic implementation of PSO [27].

IV. METHODOLOGY

This section describes the metodology,the system model and the experimentation of the study. In order to meet the goals of the study and answer the research questions, the following steps were followed.

In the first step: Create a network scenario based on the Ethio telecom backbone network and then the simulation is performed using GNS3.

In the second step: For the created scenario, BGP was configured so that the MRAI timer values were updated. The values start with 0 seconds for each MRAI and then increase by 1-second intervals until 600 was reached.

In the third step: The node failure event closer to the destination of the traffic with 600 different MRAI values produced 600 convergence time and 600 update messages results.

In the fourth step: This data was gathered to train the neuro-fuzzy system to provide the model.

In the fifth step: The optimization algorithms are applied to the model from the training system to solve the optimization problem then, evaluate the algorithms based on different metrics such as convergence rate and computational complexity such as Time elapsed and Memory usage.

In the sixth step: Based on the results, obtained from step five, the optimum value of the MRAI timer is selected from the best algorithm. Finally, BGP with the default MRAI value which is five seconds, and optimized MRAI value will be compared in terms of convergence time and packet loss.

The figure below depicts the framework of the thesis work from data collection to model experimentation and evaluation.

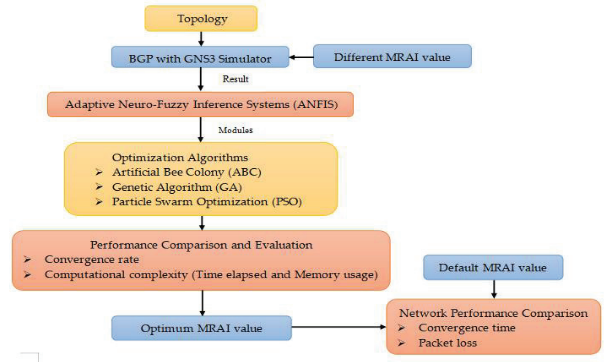


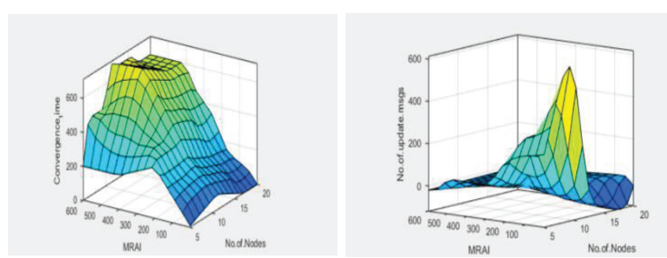
Fig. 4. System model for optimum MRAI value

V. RESULTS AND DISCUSSION

This section presented all the results obtained from testing and evaluating the selected three optimization algorithms to solve BGP slow convergence problem. Furthermore, this section compares the models using different performance metrics. The results of the tests were discussed based on this experimental analysis.

A. ANFIS

After data is generated and collected, the dataset has been ready for model training purposes. This implementation was done with the MATLAB tool. The key goal of this section is to learn and decide on the nature of BGP convergence time and the number of update messages. The intelligent system modeling tool used to accomplish this purpose is an ANFIS. The inputs to ANFIS are the average degree, the average diameter, number of nodes, and minimum route advertisement interval. As a result of the implementation of this step, the model for the convergence time and the number of update messages is given below.



(a) (b)

Fig. 5. Model of First-Order ANFIS for (a) Convergence time, (b) Number of update messages.

B. Optimization algorithms

Once the output of the ANFIS is acquired, optimization algorithms were applied to the model from the training system. The algorithms are used to optimize the parameters of the model, which means the MRAI were tuned using three optimization algorithms to minimize the convergence time without affecting the number of update messages. The implementation has been performed using the MATLAB tool.

To evaluate the performance of each model's parameters such as the rate of convergence against the number of execution to achieve its known best value; the average running time which is necessary to finish the algorithm and the average memory usage required to run the operations of the algorithm are selected. These evaluation parameters are the most significant factors for optimization algorithms to be recognized as the best models for the BGP slow convergence problem. Each approach was subjected to fifty independent trial runs with 100 iterations.

Fig.6 shows the graphical representation of the convergence rate of the optimization model's performance. According to this, the PSO algorithm has a consistent solution and faster convergence speed towards the global optimum solution followed by GA. Whereas ABC has the lowest rate of convergence as compared to GA and PSO.

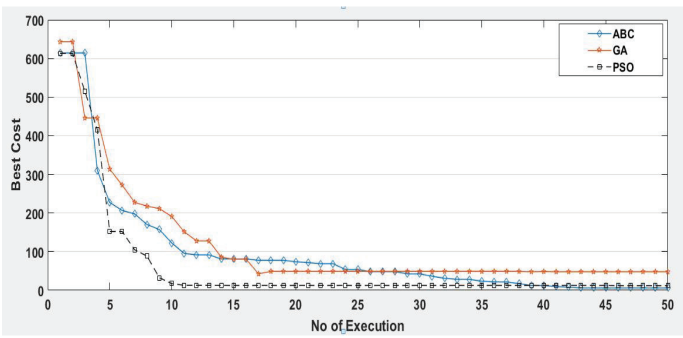


Fig. 6. Comparison of convergence rate

The time taken to complete ABC, GA, and PSO for various iterations in a generalized normalized environment with the same processor and memory was compared. It is necessary to note that the machine abilities and the implementation of the optimization techniques affect the execution time. On the other hand, memory usage refers to how much memory an algorithm consumes in order to finish the searching process. The use of a high memory implies that the system memory is being wasted. Table 1 presents the evaluation results of ABC, GA, and PSO algorithms in terms of average time and average space consumption for 50 independent executions with Iteration 100. Accordingly, the computational time and the space taken by the ABC algorithm to search for the global optimum are 8.95 sec and 777.6 KB respectively. Whereas GA consumes 5.84-sec running time and 2635.88 KB memory usage. The computational time and space required by the PSO algorithm to perform its operation are 3.96 sec and 803.12 KB respectively.

TABLE I. PERFORMANCE COMPARISON OF SELECTED ALGORITHMS

Selected Algorithm	Memory usage (KB)	Running time (sec)
ABC	777.6	8.95
GA	2635.88	5.84
PSO	803.12	3.96

As shown in the above result, the PSO algorithm took minimum execution time, whereas, in the ABC optimization model, the maximum time is taken to reach an optimal solution. The GA models presented a comparatively medium value of running time. The following figure illustrates the time complexity of each model at the same condition.

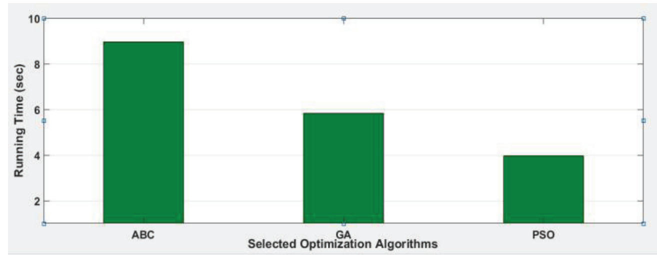


Fig. 7. Comparison of time complexity

On the other hand, the space that GA occupied is much higher than ABC and PSO. ABC and PSO use the lowest memory usage than GA but PSO have a comparably large amount of memory as compared to ABC. Figure 9 reflects the graphical representation of memory usage by each model at the same task.

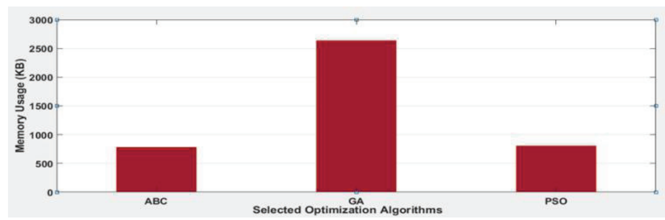


Fig. 8. Comparison of space complexity

Finally, by observing the simulation studies and results, except the memory consumption, PSO achieves the best results in all evaluation metrics than the others two techniques. It is successful in obtaining a near-optimal solution in a short amount of time and at a short convergence rate. As a result, it can be concluded that the model which is developed with the PSO optimization approach is considered as the selected working model for BGP slow convergence problem. Accordingly, the optimal MRAI is selected from the PSO algorithm. The PSO determined that two seconds was the best value for the MRAI because the convergence time was reduced with regardless of affecting the number of update messages. Based on this, Figure 10 depicts the result for the network with 17 nodes. One of the nodes failed after 2 minutes, and the BGP protocol was required to converge. The default BGP with the default MRAI value (five seconds) took 29 seconds before it converges.However,the optimized BGP with the recommended MRAI value (two seconds) reached 13 seconds, indicating that the optimized BGP converged faster than the default, the delay was minimized.

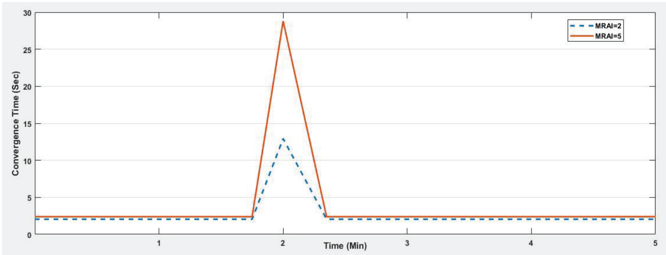


Fig. 9. Comparison between convergence time with default and optimum MRAI

As shown in Figure below, there is no packet loss in both scenarios when the nodes are not disconnected but, in both cases, the packet drop increases very fast when the node is being failed. The performance of optimized BGP is much better than that of default BGP.

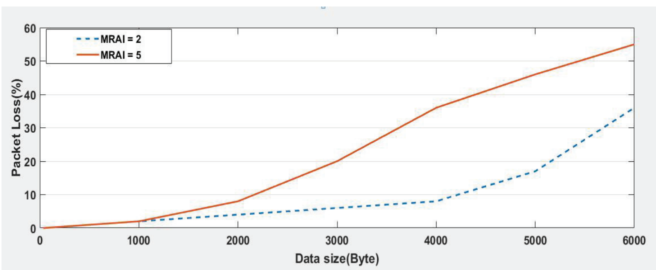


Fig. 10. Comparison of packet loss for default and optimized BGP

VI. CONCLUSION AND FUTURE WORK

The following conclusions can be drawn from the study and simulation results:

- The comparative results of the three algorithms show that PSO outperformed the other two algorithms in obtaining optimum value.
- The MRAI timer affects convergence time and packet loss. This indicates as it can be optimized to improve the convergence time and packet loss and found that two-second as an optimized MRAI timer.
- Compared to the default MRAI timer, the optimized MRAI improves the delay in convergence up to 55%.
- At the same node failure event optimized MRAI reduces packet loss by 19%.
- Any service provider, including Ethio telecom, can implement the optimum MRAI timer in BGP configuration to improve the quality of service for the company's customers and overall network performance by speeding up packet rerouting, reducing packet loss and delay in the event of failure.

The following areas are recommended for future work.

- More investigations on the combination and optimization of other BGP timers such as KEEPALIVE, HOLD timer with MRAI timer which enhances the overall network performance can be done.
- Future research may focus on the hybridization of the aforementioned algorithms to improve the search ability of PSO which detects the global optimum value of BGP parameters (mentioned in the above) for slow convergence problems in short running time and low memory usage.

ACKNOWLEDGMENT

The author of this paper acknowledges the support given by Ethio Telecom and School of Electrical and Computer Engineering, Addis Ababa Institute of Technology.

REFERENCES

[1] R.Bennesby and E. Mota, "A survey on approaches to reduce inter-domain routing convergence delay in the Internet," *IEEE Communications Surveys & Tutorials*, Vol 19, Issue 4, 2017.

[2] R. Paul, "Performance Evaluation of Border Gateway Protocol with Route Flap Damping and Routing Policies," *Thesis, Library, and Archives Canada Bibliothèque et Archives Canada*, Ottawa, 2014.

[3] J. Park, "Understanding the Impact of Internal BGP Route Reflection," *Ph.D. dissertation*, University of California,2011.

[4] C. Labovitz, et.al., "Delayed Internet routing convergence," *In IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 293-306, 2001.

[5] S. Deshpande and B. Sikdar, "On the impact of route processing and MRAI timers on BGP convergence times," *IEEE Global Telecommunications Conference (GLOBECOM)*, Vol.2, pp. 1147-1151, Dallas, TX, 2004.

[6] A.Fabrikant, et.al., "There's something about MRAI: Timing diversity exponentially worsens BGP convergence," *Proc. of INFOCOM*, Shanghai, iBGP, China, 2011.

[7] E. Rekhter, et.al., "A Border Gateway Protocol 4 (BGP-4)," RFC 4271.1970. [Online]. Available: https://www.https://tools.ietf.org/html/rfc4271. [Accessed: 09-Dec-2020].

[8] N. Laskovic, "BGP with an adaptive minimal route advertisement interval," *Simon Fraser University*, Canada, 2007.

[9] B Ricardo, "DeepBGP: A Machine Learning Solution to reduce BGP Routing Convergence Time by Fine-Tuning MRAI," *Universidade Federal do Amazonas*., Doctor of Informatics, Brazil, 2019.

[10] H. Alshamrani, "Detecting IP prefix hijack events using BGP activity and AS connectivity analysis," *University of Plymouth Research Theses*, 2017.

[11] J. Qiu, F. Wang, and L. Gao, "BGP rerouting solutions for transient routing failures and loops," *In Military Communications Conference*, 2006. MILCOM 2006. IEEE, pp. 1–7, 2006.

[12] J. John, E. Katz-Bassett, A. Krishnamurthy, T. Anderson, and A. Venkataramani, "Consensus routing: The internet as a distributed system," *In Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'08, (Berkeley, CA, USA), USENIX Association, pp. 351–364, 2008.

[13] B. Wang, "The research of BGP convergence time," *2011 6th IEEE Joint International Information Technology and Artificial Intelligence Conference*, pp.354-357, 2011.

[14] S. Agarwal, C. Chuah, S. Bhattacharyya, and C. Diot, "The impact of BGP dynamics on intra-domain traffic," *In Proceedings of ACM SIGMETRICS*, vol. 32, (New York, NY, USA), ACM, June 2004.

[15] S.Shukla and M. Kumar, "Improving Convergence in iBGP Route Reflectors," *Lecture Notes in Electrical Engineering Advances in Electronics, Communication, and Computing*, pp. 381–388, Springer, Singapore, 2017.

[16] S.Shukla and M.Kumar, "Optimized MRAI Timers for Border Gateway Protocol in Large Networks," *International Journal of Distributed Systems and Technologies*, vol. 10, no. 4, pp. 31–44, 2019.

[17] A.Nykvist, "Convergence Issues in Inter-Domain Routing and Connectivity Modeling of Ad hoc Networks," *Luleå University of Technology*, Department of Computer Science and Electrical Engineering, 2006.

[18] A. Ahmed and S. M. Shah, "Application of adaptive neuro-fuzzy inference system (ANFIS) to estimate the biochemical oxygen demand (BOD) of Surma River," *Journal of King Saud University - Engineering Sciences*, vol. 29, no. 3, pp. 237–243, 2017.

[19] D. Karaboga and E. Kaya, "Adaptive network-based fuzzy inference system (ANFIS) training approach: a comprehensive survey," *Artificial Intelligence Review,* vol. 52, no. 4, pp. 2263–2293, 2018.

[20] M. Buragohain, "Adaptive Network-based Fuzzy Inference System(ANFIS) as a Tool for System Identification with Special Emphasis on Training Data Minimization," *Thesis, Indian Institute of Technology Guwahati*, Guwahati, 2009.

[21] S. Kumar, "Nature-Inspired Computation Methods and Their Application in Function Optimization," *Thesis, Faculty of Engineering and Technology*, Jagannath University,2014.

[22] A. Wahab, M. Nadhir, S. Nefti-Meziani, and A. Atyabi. "A comprehensive review of swarm optimization algorithms," *PLoS one vol .10*, no.5, pp. 1–36., Catalin Buiu, Politehnica University of Bucharest, ROMANIA, May 2015.

[23] A.Hussein and R.Khalid, "A Comparative Study of Swarm Intelligence-Based Optimization Algorithms in WSN," *Asian Journal of Engineering and Applied Technology*, vol. 8, no.3, pp. 1-7, 2019.

[24] M. Rani, "Multi-objective optimization of PID controller parameters using genetic algorithm," *Thesis, Universiti Teknologi Malaysia*, December 2012.

[25] S. Katoch, S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future. Multimedia Tools and Applications," *Computer Science and Engineering Department, National Institute of Technology*, Hamirpur, India, 2020.

[26] Y. Ting, et al. "Swarm intelligence optimization algorithms and their application," *Association for Information Systems AIS Electronic Library (AISeL)*, WHICEB 2015 Proc 3, pp. 201-206,2015.

[27] F. Bergh, "An analysis of particle swarm optimizers," *Thesis, University of Pretoria, Pretoria*, 2001.