Journal of Theoretical and Applied Information Technology
28th February 2017. Vol.95. No 4

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCED BORDER GATEWAY PROTOCOL IN NS- 2 BY ADDING THE HOT POTATO FUNCTIONALITY BASED ON REAL NETWORK

SAMI ABBAS NAGAR, SULAIMAN MOHD NOR, MOHAMED SAAD BOBA
Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Malaysia
E-mail: elnajarsami@yahoo.com, s_mohdnor@utm.my, bobba802013@gmail.com

ABSTRACT

The rapid growth of the Internet has made the issue of ensuring reliability and redundancy a big challenge. Studies of these issues using Traffic Engineering and simulation have been extensively done. There has been substantial interest from researchers in the development and contribution of modules in NS-2. Most studies have not taken into account real traffic parameters in their simulation models. Also, there is no comprehensive model consisting of Border Gateway Protocol (BGP) and Hot Potato (HP) routing in the NS-2 network simulator based on real networks. In this paper, Integrated Model is introduced consisting of HP algorithm and BGP integrated into the NS-2 network simulator. The integrated model is then used to simulate the infrastructure of a real production network using actual captured traffic data parameters. The network is modeled with a baseline topology where 5 main nodes were connected together, with redundant links for some nodes. The simulations were repeated for link failures. HP helps in improving the node which experiences a link failure to select shorter distance route to egress router. In the case of a link failure, HP switching time between the links is 0.05 seconds. The integrated model performance was evaluated by comparing trace file before and after link failure or by adding nodes (up to 32). The parameters used for comparison are the packets loss, delay and throughput. The integrated model error percentage obtained for packets loss is 0.025%, delay 0.013% and throughput 0.003%.

Keywords: *BGP, HP, NS-2, Packet loss, Delay, Throughput.*

1. INTRODUCTION

In the network research area, it is time consuming and costly to deploy a complete experimental testbed containing multiple networked components such as computers, routers and data links to validate and verify a specific network algorithm or a certain network protocols.

The use of network simulators in these cases will save the time and money in accomplishing this task. Network simulators are also mostly useful in allowing the network researchers and designers to test new or to modify existing networking protocols in a controlled and reliable manner [1].

Network Simulator is a pure event based simulator and can be of two types [2]

- Discrete Event Simulator.
- Continuous Event Simulator.

Generally network simulators are discrete event simulators. As shown in Table 1, network simulators can be categorized according to commercial and open source based. Also web technologies in modeling and simulation recently are available [27]. Figure1 shows the extensive use of NS-2 compared to other simulation or development tools [5].

Table 1: Network Simulator

Type	Network simulator
Commercial	OPNET, QualNet
Open source	NS-2, NS3, OMneT++ , SSFNet, J-Sim

NS-2 simulator contains modules for many network components such as routing, transport layer protocol and application. NS-2 used to investigate the network performance such as congestion or link failure [3].

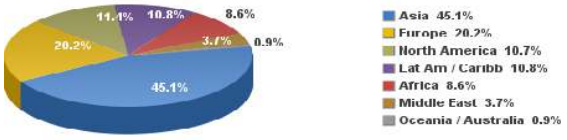

Figure 1: Simulator Usage

Journal of Theoretical and Applied Information Technology
28th February 2017. Vol.95. No 4

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Border Gateway Protocol (BGP) exchanges the information with neighbors (peers) in different AS. BGP has attributes which may not allow the neighbor to communicate with BGP node. BGP is divided in two parts based on routing functionality. EBPG (exterior) which is used to link the egress point together in different AS's. IBGP (interior) used for linking the nodes Interior Gateway Protocol (IGP) together in the same AS's. Figure 2 describes BGP types.

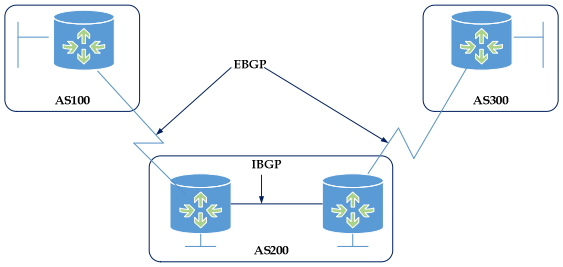

Figure 2: EBGP and IBGP

Hot Potato (HP) is one of BGP decision selection process. HP works when BGP runs in IBGP mode. The node in the same AS has two routing tables, one for BGP and another for an EIGRP routing table as shown in Figure 3. HP selects the smaller distance to the next hop (egress point) which it learns from the node EIGRP routing table.

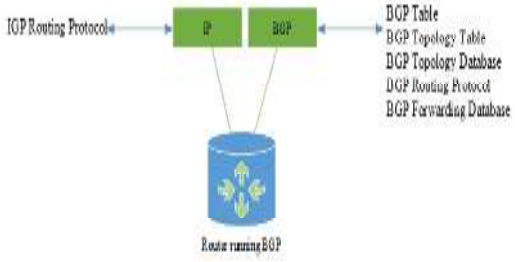

Figure 3: The Router Is Running BGP and IGP

HP will provide BGP with parameters to make one of the BGP decision process. BGP will establish a TCP connection on TCP port 179 between two ASes, then the updated messages send between the ASes which include information about BGP metrics (path attribute). These attributes include well-known mandatory attribute such as AS-path, Next-hop and Origin; well-known discretionary attribute such as local preference and atomic aggregate; optional transitive attribute such as Aggregator and community, optional non-transitive attribute such as Multiexit-discriminator (MED).

The steps involved in the BGP decision process are illustrated as follows:

1. Ignore if egress point unreachable.
2. Highest local preference.
3. Lowest AS path length.
4. Lowest origin type.
5. Lowest MED (with same next-hop AS)
6. EBGP - learned over IBGP – learned.
7. Lowest IGP distance to egress point ("Hot Potato").
8. Vendor – dependent tie break.

NS-BGP patch for NS-2 driven from SSF.OS.BGP4 is implemented in SSFNet [10][11]. All the features which are available in the SSF.OS.BGP4 are ported to NS-2 version 2.27 by Tony [12]. The reference in this work for both network simulators is RFC 1771. NS-BGP for NS-2 version 2.34 also is available. IBGP has two main routing mechanism, route reflectors and confederations. SSFNet and NS-BGP simulators are only providing the Route Reflection (RR) in IBGP routing mechanism.

The motivation for this research is geared towards solving the Corporation network of Sudan ISP. As at the moment, this research only involves simulation environment. Thus, it is important to implement these algorithms in routers with some provisions, so that automatic update of the input parameters for the router and the algorithm could be achieved. This has to be seriously considered in second part of this research.

This paper aims to demonstrate how use the real input parameters readily available can be used instead of NS-2 available traffic generators [13]. These values are used as input parameters in the simulation model representing an actual production network. Integrated Model is introduced consisting of HP algorithm and BGP integrated into the NS-2 network simulator. Validation is performed by comparing the metrics throughput, delay and packet loss of the simulated network before and after uses HP.

2. LITERATURE REVIEW

There are some limitations to network simulation that even NS3 cannot overcome. One of them is credibility. This will always be an issue, because it is clearly impossible to guarantee flawless real world behavior of a simulation. One

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

approach to partially solve this problem could be a far more detailed formalization of the validation process. To know the limitation of upper layer functionality between NS-2 and NS3 it needs a simple comparison instead of validation [9].

In NS-2 the packet transmission time is equal to packet size divided by bandwidth for each link, and transmission delay or packetization delay or store and forward delay depend on which queue mechanism used such as DropTail, RED and FQ. (See code fragment as in Figure 4) [3]. However, in the real world systems, packet transmission time calculates for each link affected by a number of elements such as:

1) Source CPU time.
2) Link transmission packet time.
3) Link propagation time.
4) Intermediate forwarding time.
5) Destination CPU time.

The values of Source CPU time, link propagation time, destination CPU time and intermediate forwarding time are significantly very smaller than one (<<1) and thus can neglect.

```
  # Dump the queueing delay on the
n0->n1 link
  #  to  stdout  every  second  of
simulation time.
  #
  proc dump {link interval} {
       global ns integ
       $ns  at  [expr  [$ns  now]  +
$interval] "dump $link $interval"
       set delay [expr 8 * [$integ
set  sum_]  /  [[$link  link]  set
bandwidth_]]
       puts         "[$ns         now]
delay=$delay"
  }
```

Figure 4: Code Fragment In NS-2

"P. R. Gundalwar1" [14], used OPNET Academic IT Guru Edition 9.1 for network simulation study of BGP as the routing protocol. In their paper, they discussed the comparative results for BGP simple routing policy needed in any network administration for effective network utilization.

2.1 Network Simulator 2 (NS-2)

Network simulator-2 (NS-2) is an open source discrete event simulation tool used for simulating Internet protocol (IP) networks [15]. NS-2 is a part of the VINT project (Virtual Inter Network Testbed) is developed in C++. It uses an IU OTCL interpreter. The NS-2 simulator uses TCL [7] as a front-end interpreter and C++ as the back end network simulation engine. Through this language, the user can describe conditional parameters of the simulation. The user can also create new objects in C++ and use them in NS-2 by instantiations with OTCL. Here, the two languages C++and OTCL have both very close hierarchies to each other. [3][4][6].

The traffic agents in NS-2 are used to generate different types of traffic based on real traffic network. A modified sniffer program was used to capture real traffic data from a production network and output to a traffic text file. This traffic test file is then used by the NS-2 traffic agents in the simulated network model representing the real production network. To validate that the simulation works accurately, the output trace file of NS-2 after simulation is compared with the traffic generated at the destinations of the production network. Error percentage and t-test data analyses were conducted. It was found, based on comparisons that the difference was nearly zero for traffic captured for duration of around 15 minutes simultaneously generated from four different production nodes. [13]. NS-2 is one of the best open source simulators of the researchers, because it easy to adding your model or work to NS-2, such as an IEEE802.11 MAC approaches in NS-2 [22], and Implementation of TCP_Reno Algorithm in the ns2 [25].

2.1.1 Communication entity in NS-2

The node (communicating entity) is the basic element of our model. A node in NS-2 is a class defined in the OTCL which has three entities containing: the Classifier, the Link and the Agent [3]. Figure 5 show the relation between the communications entities.

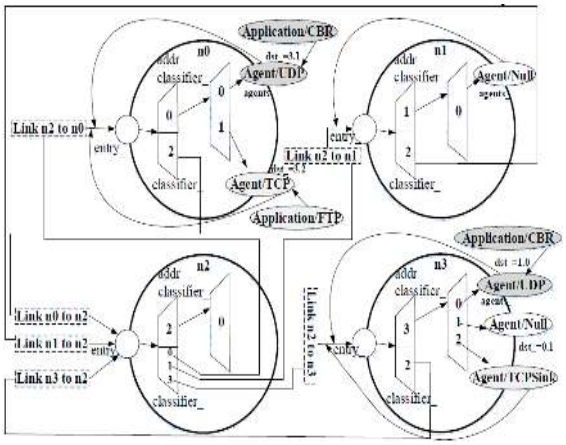

Figure 5: The Existing Entities In A Node And The Links Between Entities

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.1.2 Unicast NS-2 structure units

The unicast consists of two parts, a control (route computations) and forwarding (classifying and forwarding). In Figure 6 the blue ellipse shows the forwarding elements and the red shape shows the control elements. This structure is used by dynamic routing in NS-2.

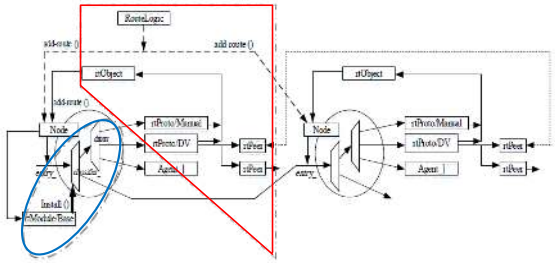

Figure 6: Unicast Structure In NS-2

2.2 AWK Language

The AWK utility is an interpreted programming language typically used as a data extraction and reporting tool. It is a standard feature of most Unix-like operating systems. AWK Scripts are very good in processing the data from the log (trace files) which we get from NS-2. It is necessary for the researchers to know the throughput of the network, packet delivery ratio, calculating the sound, received, dropped packets, and average end to end delay [8].

For NS-2 to be representing the real network, NS-2 must use real time which means the time of simulation equal to real time of production network of dataset interval and use the real parameters such as link information to perform the simulation of production network. There are some assumptions to be made, such as the packet time equals the total packets number divided by interval time. From this value, inference on the interval time between packets during the execution of simulation can be made. The packet size equals to the total packet length divided by total numbers of packets, the value inferred from this equation is used as constant packet size. In both methods mentioned above there is an error percentage (Liu et al., 2009). Some researchers uses the packet off/on time to send only one packet in actual time in the real network to make the send error percentage for packet 0%, but uses the constant packet size not the actual packet length.

2.3 Border Gateway Protocol and Hot Potato

Large Internet Service Providers (ISPs) experience millions of BGP routing changes a day [16]. In this paper, they discuss the impact of BGP routing changes in the flow of traffic, summarizing and reconciling the results from six measurement studies of the Sprint and AT&T backbone networks.

The researcher uses BGP to improve ISP routing, such as link failure, and packet loss. Likewise, delay and throughput metrics are used for benchmarking the results after the enhancement against the metrics result before the experiments are performed [23] [16] [24] [28] [29] [30]. These results depend on different inputs (parameters) such as dataset, times between events in seconds (link down/up or add/remove node) and the system specifications used in the experiment [31].

The original BGP design requires that all BGP speaks within an autonomous system directly connected with each other to create a full mesh, and BGP update messages propagated directly to connect neighbors only [17]. This requirement leads to BGP session scalability problems in networks with large numbers of BGP routers.

Hot Potato (HP) Routing is the practice of passing traffic off to another autonomous system as quickly as possible, thus using their network for wide-area transit [18]. They studied the relation between BGP, and with any IGP protocols. HP is responsible to select the smaller distance to the egress point which supports BGP to select the path. This relation between BGP and HP is only done analytically. This is the model proposed in [19]. HP adds the advantage to BGP when deciding to select the specific route by reducing the overheads such as resource consuming inside an AS [20].

Hot Potato is a very valuable algorithm for BGP particularly when running in IBGP mode for selecting egress points that are very close. However, this selection does not carry considerable link utilization. Thus, HP could constitute some delay to BGP, such as selecting the closer egress point which has the congestions. There are on-going researches in solving the problems of combing HP routing mechanism with egress selection method. However, concerted efforts need to be geared towards using it in a simulation

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

environment, wherein their impacts of the HP could be determined as compared to real environment. To date, there is no existing network simulator that supports HP functionality [19] [21] [32] [33] [34] reported that Hot Potato routing can be a significant source of BGP, since it updates can lag for 60 seconds or more behind the intra-domain event (refer to Table 2 for type of IGP protocol used). Also, the number of BGP path changes triggered by Hot Potato routing has a nearly uniform distribution across destination prefixes, and the fraction of BGP messages triggered by intra-domain changes varies significantly across the time and router locations. They show that Hot Potato routing changes lead to longer delays in forwarding-plane convergence, shifts in the flow of traffic to neighboring domains, extra externally-visible BGP update messages, and inaccuracies in Internet performance measurements updates.

Table 2: Comparison of protocols, ports, reliability, and default timers

	RIP V1—Routing Information Protocol	IGRP—Interior Gateway Routing Protocol	EIGRP—Enhanced Interior Gateway Routing Protocol	OSPF V2—Open Shortest Path First	DGP V1-Border Gateway protocol
Protocol Number	-	9	88	89	-
Port Number	UDP 520		-	-	TCP 179
Update Reliability	Best-effort delivery	Best-effort delivery	1-to-1 Window	1-to-1 Window	Uses TCP windowing
Update Frequency	30 sec plus triggered	90 sec plus triggered	Triggered	Triggered also, Link-state advertisements (LSAs) flooded every 30 min	Triggered
Hello frequency	-	-	60 sec for multipoint T1 or less 5 sec for others	30 sec - no broadcast multicast (NBMA) 10 sec - others	60 sec
Other Timers	Hold and invalid timers - 180 sec Flush - 240 sec	Hold timers - 280 sec Invalid timers - 270 sec Flush timers - 630 sec	Hold timers - 280 sec for multipoint T1 or less 15 sec for others (SiaActive interval)	Dead-120 sec NBMA-40 sec others (4xHello interval)	Hold - 180 sec

2.3.1 Unicast BGP in NS-2 structure

To apply BGP in NS-2 (NS-BGP), the unicast structure must be modified according to BGP functionality added by Tony [12]. Some components are replaced with a new one, such as classifier IPv4 instead of the basic classifier. The routing, BGP algorithm (rtProtoBGP) is responsible for all BGP activities. Also new components, such as TcpSocket are added to the unicast structure to support the user data transmission. Figure 7 shows the NS-BGP unicast structure.

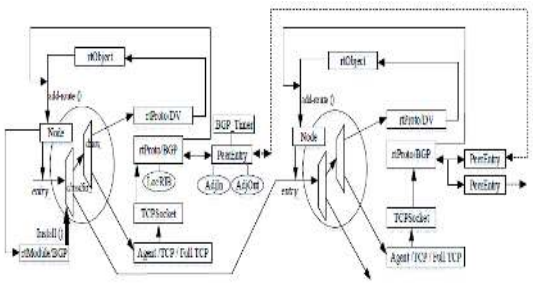

Figure 7: BGP Unicast Structure In NS-2

There is lack of researches on the use of Hot Potato routing functionality in BGP in simulators network [11] [19] [31]. BGP patch can be found in many network simulators. However, even if BGP exists in simulators, there is no interconnection between BGP and any IPG protocols to support BGP selection decision process [29]. [31] Demonstrates how BGP running without any update information, is used to select the alternative path. With an experimental time in 20 seconds and one link down for 4 seconds, the researcher obtains the following for the three performance metrics: error percentage is 22% for packet losses, 12% for average delay and 22% for throughput.

2.4 Validation and Verification

To simulate the production network there are some steps should be followed to verify and validate the accuracy of simulation [13,26]. Different protocols are then evaluated based on measures such as the packet loss rate, the overhead introduced by the routing protocol, end-to-end packet delays, network throughput, and ability to scale.

The problem of the Corporation is to move from the current network configuration which uses static routing protocol to future state network that is based on Traffic Engineering (TE). Thus, there is a need to minimize the real experimental risks as much as possible by using a suitable network simulator NS-2 is normally used to simulate the real network based on the topology and link probabilities. However, networks parameters representing the real network traffic are not used in NS-2. The error percentage observables with BGP simulated in NS-28 with three matrices are 22% for packet losses, 12% average delay and 22% throughput. There is lack of such comprehensive model consisting of BGP and HP in NS-2 based on real network from previous studies and analysis. Previous work such those found in [21], only HP functionality were tested in BGP and which were used to generate the traffic matrix in order to study the network impacts.

3. DESIGN AND IMPLEMENTION OF AN INTEGRATED SIMULATION MODEL

This design discusses on the research methodology used in this paper. The paper methodology considers the organization of the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

research design and procedure, and describes the way forward towards achieving the paper objectives. The detailed analyses of the methodology are explained briefly in the subsequent modules.

HP function added to BGP in NS-2model is near to realistic to generate the traffic from giving network text file [12]. The goals of this addition are:

1) To practice the real parameters to construct the network such as the number of nodes, distance between nodes, topology, link details, queue mechanism etc.
2) To handle the traffic in the network when the link fails.

The ultimate goal of the research is to propose network architecture with inherent practical dynamic routing which takes into consideration the various scenarios such as link failures which will impact the performance of the network. Practically, this is done via a simulation environment to study the experimental impact before implementing the algorithm in the real network. This work is divided in three sub goals:

1) Build HP pseudo algorithm and validate by running the conceptual validation based on HP advantages. Figure 8 shows the flow steps to validate HP algorithm. The validation is based on evaluating the benefit of HP based on its features. HP has two important features, automatically adapting to the network topology changes and reducing the resource consumption it learned from EIGRP. The algorithm must pass the two beneficial features which are used to validate the HP algorithm.

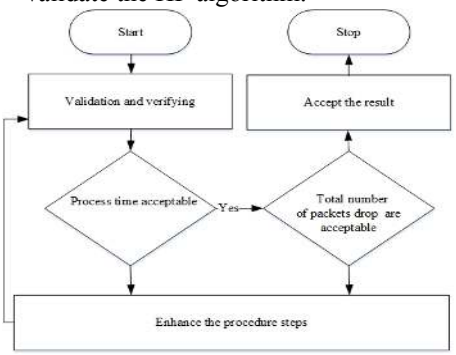

Figure 8: HP Validation Steps

2) Add HP functionality to BGP in NS-2 to handles the link failures impact by modifying the BGP unicast structure in NS-2. The network simulation model is itself a representation of the production network [13]. Figure 9 gives an overview of this framework.

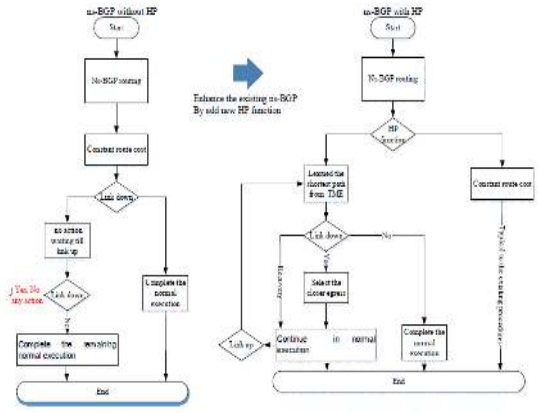

Figure 9: Shows The HP New Function Added To ns-BGP

3) Runs different Scenarios for the real production network in NS-2 using the real traffic, which captured by sniffer software from the aggregation router.

3.1 Build the HP algorithm

In the follow subsection we will explain the HP algorithm.

3.1.1 Algorithm description

HP algorithm learns the distance from the EIGRP routing table. The algorithm is coded using C++/C language.

3.1.2 Data and assumption

The network topology under study is a dual-homed network as shown in Figure 10. The network consists of two AS's, AS0 consists of 4 BGP nodes (A, B, C and D), while AS1 has only one BGP node (E). The node B in AS0 is running iBGP with HP algorithm. Node B is linked to two egress points C and D, to reach AS1. TCP data which have constant packet size 1040Kbps is sent between node A and node E.

In this work we make certain assumptions which are as follows:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1) BGP connection has been established when executing HP algorithm.
2) EIGRP sends triggered update message every 45 seconds.
3) When the link down/up happens, The HP algorithm needs 0.001 seconds to select the alternative path after getting the update from EIGRP.

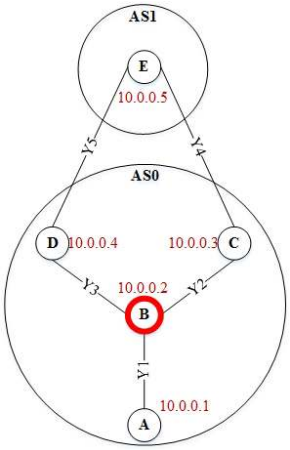

Figure 10: The Dual-Homed Network Topology

3.1.3 Pseudo code of the algorithm

The algorithm loads all the distances obtained from EIGRP into the hash table, where the program is written in C++. HP selects the egress point of searching the hash table and selecting the smaller distance for the specific link between the two nodes. Our algorithm has the following steps when executed:

1) Start transmits the UDP between AS0 and AS1.
2) Link failure (Link Down) when packet ID = the input packet ID.
3) HP functionality is searching the smaller distance in the hash table.
4) Select the smaller distance to the egress point.
5) Use the selected path (alternative path) for transmitting the UDP between AS0 and AS1.
6) Link recovers (Link up) when packet ID = input packet ID.
7) Repeat step 3, 4 and 5.
8) The Trace file contains all the network traffic.

3.1.4 Validation

HP algorithm was validated concepts. The benefits of HP presence are verified.

3.1.5 Implementation

HP algorithm is used for updating the BGP routing from EIGRP. This is further explained below with two scenarios. The first example is to know the impact of HP when HP functionality is not used. In the second example, the impact of HP when HP is updated from EIGRP is examined. Finally, the results of the two scenarios, based on the packet loss and selecting the alternative path were compared.

3.1.5.1 Scenario I: HP functionality not enabled

In this first scenario, HP functionality is not in used. The algorithm input parameters are shown in Table 3. These parameters are user defined. The total number of packets should be transmitted, link down (Y) at packet number (Packet ID (down)) and the link (Y) up at packet number (packet ID (up)).

Table 3: The Program Input Parameters Assigned By The User

Input	Integer value
Total number of packets	240
Link down (Y?)	3
Packet ID (down)	50
Packet ID (up)	75

Figure 11 shows the trace file which consists of sign, source, destination, TCP protocol and the packet ID. The second row in the trace file shows the link failure between node B and C when the packet ID equal to 50, which means node E cannot receive the packet from node A via node B and C, although there is another link between B and E via D as shown in Figure . 11. This link can be used as an alternative link between B and E as shown in the Figure 9. HP functionality is not in use so the alternative link is not being addressed by node B to handle the traffic between node B and node E. The link between node B and C gets recovered when the packet ID equals to 75 and node E starts receiving packets from node A. This means 25 packets are sent from the source (Node A) to destination (Node E) via node B, and these packets are not received at the destination.

3.1.5.2 Scenario II: HP functionality enabled

In this second scenario, HP learns the distance for all egress points from the EIGRP (hash

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

table) which send the update message every 45 seconds and uses the same parameters in Table 3.

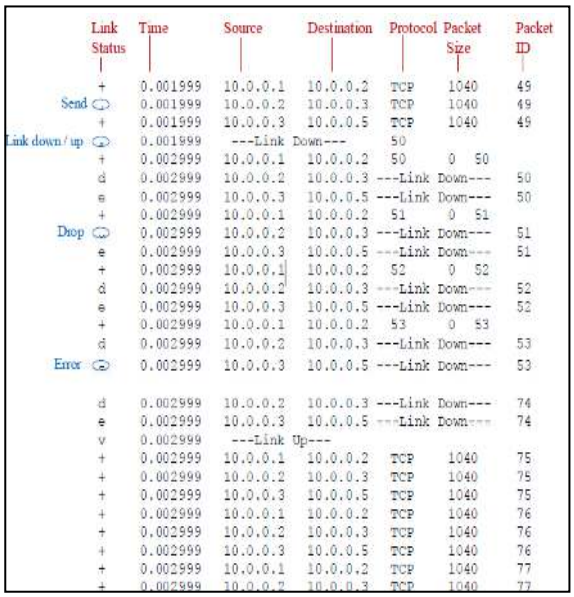

Figure 11: BGP Without HP Function

In Figure 12, HP functionality is in used so the alternative link is addressed by node B within the switching time greater than 45 seconds. The alternative link handles the traffic between node B and node E. The link between node B and C gets recovered when the packet ID equals to 75 and node E starts receiving packets from node A. This means only 4 packets are sent from the source (Node A) to destination (Node E) via node B and these packets are not received at the destination.

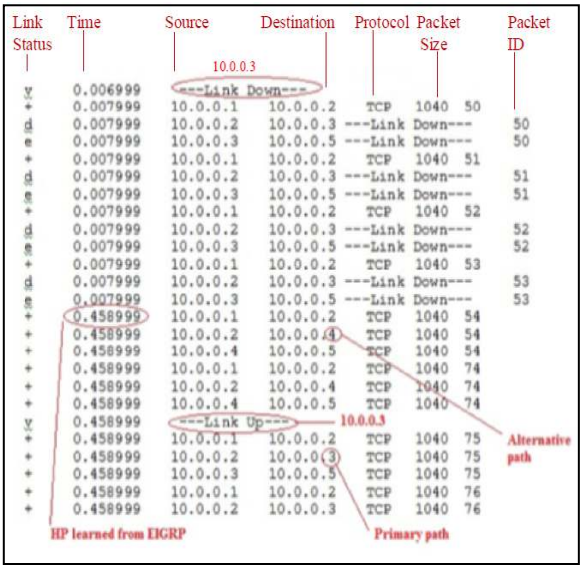

Figure 12: HP Learned From EIGRP

3.1.6 Result

Table 4 shows the comparison between the standard algorithms, HP learned from EIGRP. The comparison is based on packet loss and switching time to the alternative path. Both parameters highlight the benefits of HP. Here, the benefits in terms of the features used and represented by the parameters are used to verify the functionality of HP. The result shows HP algorithm is functioning in switching to the alternative path. HP time to search the hash table is 0.05 seconds.

Table 4: Result Comparison

Scenario	Total packet loss	Alternative time
Standard	25	= 25 packet times
HP with EIGRP	4	<45 Sec

The time, which is shown in the Table 3, is not the actual time. The actual time has to consider the router processing time as explained in HP scenario section.

3.2 BGP Integrated with NS-2 Simulators (ns-BGP2.0)

The default parameter values used in this experiment are shown in Table 5. The first three parameters are interval values for the BGP timers. The simulation time is 40 seconds.

Table 5: Default Values Of Parameters Used In Reflection2.tcl

Parameter description	Default value
Hold-time interval	90 Sec
Keep-time interval	30 Sec
MARI	30 Sec
Jitter keep-alive interval	Yes
Jitter MARI	Yes
Jitter start-up timer interval	Yes

3.2.1 ns-BGP Validation

Feng [12] , attaches the validation TCL files for each process such as drop peer and reconnect peer.reflection2.tcl.

Three ASes (AS0, AS1 and AS2) are connected in a line. AS0 contains eight BGP routers, the others just one each. AS0 has two clusters, cluster 1000 and 2000. Cluster 1000 has two Route Reflectors (RR), n0 and n1. n2, n3 and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

n4 are Route Reflection Clients (RRC) of both n0 and n1. Cluster 2000 contains one route reflector n5, which has n6 and n7 as its route reflection clients as seen in Figure 13.

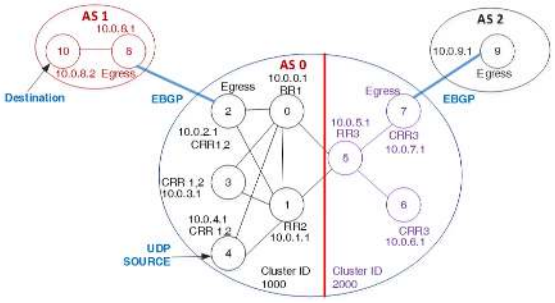

Figure 13: The Network Topology In Reflection2.tcl Validation Test File

AWK language is used to calculate the packet loss for wired network from the equation 1, the throughput from equation 2 and the average delay from equation 3.

$$PacketLoss = GeneratedPackets - ReceivedPackets \quad (1)$$

$$Throughput = \frac{Received\_data) \times 8}{DataTransimissionPeriod} \quad (2)$$

$$AverfageDelay = \frac{TotalDelay}{Count} \quad (3)$$

Here, i is the packet sequence number

$$Delay[i] = receiving\_time[i] - sending\_time[i] \quad 3(a)$$

$$Total\ DeLay = Total\ Delay + delay[i] \quad 3(b)$$

$$Count = Total\ packet\ count \quad 3(c)$$

By applying the above equations in the reflection trace file, the three parameter values were obtained as shown in Table 6.

Table 6: The Throughput, Packet Loss And Average Delay Values For Reflection2.tcl Trace File

Total of packet sends	Total of packets received	Total of packet loss	Average delay Sec	Throughput bit per Sec
79326	79326	0	0.004644183	317978.2

3.2.2 Result in Different Scenarios Applied to reflection2.tcl (Before adding HP)

This value is very small as shown in Table 7, because site 4 sends packets to site 0 and then site 0 to site 2 and site 2 to site 8 and finally site 8 to site 10. The link between site 0 and site 2 is down from 1.5 second up to 5.5 seconds. During this period only site 4 is sending packets to site 0. The packets will be discarded after the TTL time equal zero for each packet.

Figure 14 shows the comparison between the normal transmission between site 4 and site 10 and the link in between is down. The command below is used with reflection2.tcl code.

Link failure between site 0 and site 2

```
$ns rtmodel-at 1.50 down $n0 $n2
$ns rtmodel-at 5.50 up $n0 $n2
```

Table 7: The Throughput, Packet loss and average Delay values for reflection2.tcl after link failure trace file

Total of packet send	Total of packet received	Total packet loss	Average delay sec	Throughput bit per sec
17303	17119	184	0.058389346	68889

3.2.3 HP Functionality added to ns-BGP

Some assumptions are used in the HP algorithm. Firstly, it is assumed that the time needed by HP to search the hash table, which contains the 5 sites is 0.05 seconds. This parameter is set manually in HP function and not linked to the OTCL. This parameter value is set as a constant in the C++ program. This value is obtained after testing HP code to search the value from the hash table.

HP algorithm as explained earlier is added to ns-BGP by modifying the rtProtoBGP.c file and trace.c and other files which are used for route information and added route.

The steps are shown in the Figure 15 to modify the BGP for supporting HP functionality when running reflection2.tcl. There are two main conditional steps. The first step if there is any link failure when executing the simulation. The second condition relates to the BGP type selection, if it is EBGP or IBGP.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

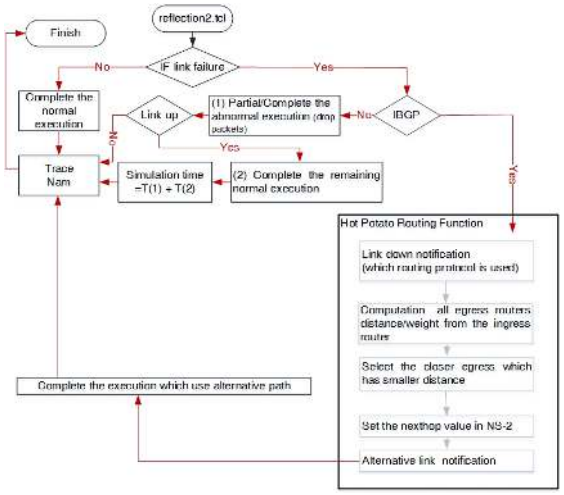

Figure 15: Hot Potato Routing Function to BGP in NS-2 with IBGP

HP function is added to the BGP routing protocol (rtProtoBGP.c). Figure 16 shows the modification on ns-BGP unicast structure. Modification of some C++ and TCL files and NS-2 reconfiguration and make has been done. This is a very important step needed to reconfigure NS-2.

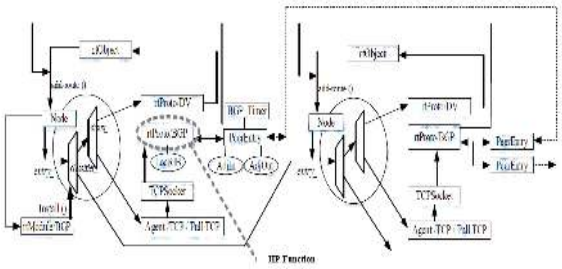

Figure 16: HP Function Added To rtProtoBGP

Figure 17 shows the detail actions when there is a link failure or link is restored. The addition or removing the node address from forwarding table is done by using the IPv4 classifier. There are two functions in the BGP routing protocols for this purpose. We add or remove the new route directly to the node classifier. Here, there is no need to update or notification messages.

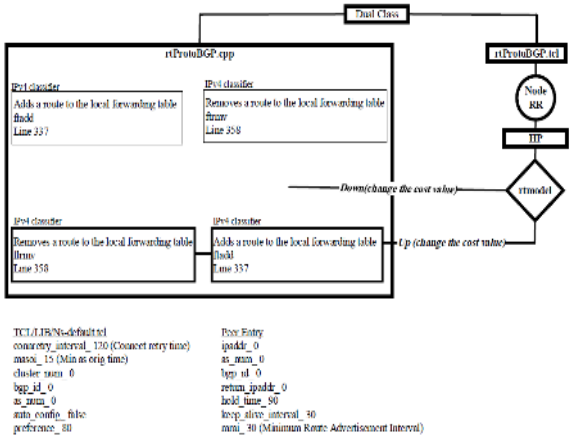

Figure 17: How HP Function Deals With Node Classifier

3.2.4 Revalidation of reflection.tcl after adding HP functionality to BGP

What happens and demonstrated in the reflection2.tcl simulation file in the case of link failure was discussed in the previous stage in section Result in Different Scenarios Applied to reflection2.tcl. The results are given in Table 5. Figure 18 shows NAM snapshot of reflection2.tcl (the original test ns-BGP TCL file for route reflection) without the "rtmodel" when the link is down for a while or the link is restored. Figure 19 shows NAM snapshot of modified reflection2.tcl by using "rtmodel". This means the link is down between site 0 and site 2 in AS0 from 1.5 seconds to 5.5 seconds.

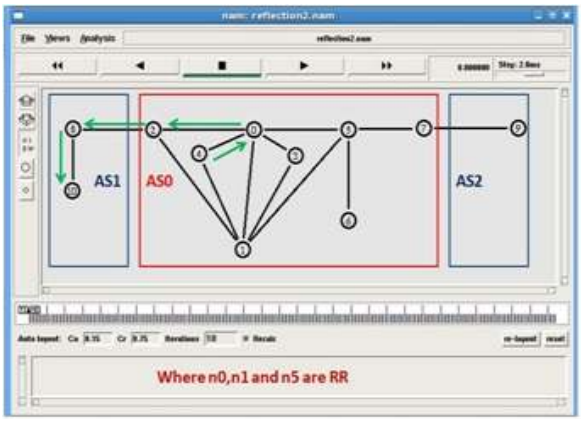

Figure 18: Normal Execution OF Reflection2.tcl

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

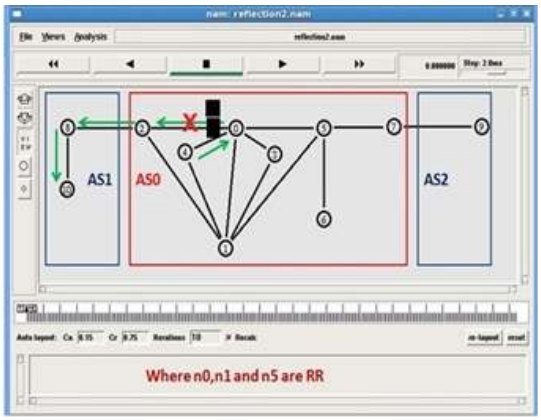

Figure 19: Execution With Link Failure of Reflection2.tcl

HP knows the link failure from "rtmodel" and immediately starts their internal process. Here there is a need to receive the update message from any IGP routing.

When the link is down, HP functionality starts to find the smaller egress point distance. This is done by doing a search in the hash table. Figure 20 shows the trace file for the complete scenario. In Figure 22, as highlighted in the green rectangle, at 1.5 seconds, HP starts to select the alternative path. HP needs 0.05 seconds to finish the searching and make the decision. As mentioned earlier, the time for the simulation is based on the laptop hardware and software specifications. Protocol such as EIGRP routing protocol.

3.2.5 Validation Procedure

Before using HP functionality, reflection2.tcl will be validated and thereafter reflection2.tcl will be re-validated after using HP functionality. The validation will compare both scenarios based on the trace files.

3.2.6 Modified reflection2.tcl

In Figure 20, node 4 in AS0 sends UDP packets via node 0 and node 2 to node 10 in AS 1 via node 8. Then the link between node 4 and node 2 via node 0 fails as seen in the black rectangle in Figure 20. The statement in this rectangle shows the modification done on BGP by adding the new HP functionality. In the statement:

```
"B 15 SAN (2014) -BGP/HP- action-
in-process-link-down 4 0"
```

The B means BGP action, 15 is the times in seconds, SAN (2014) author name, BGP/HP the HP function starting the processing and 4 0 means the link between node 4 and node 0 is down.

ns-BGP without using HP functionality is represented in the black rectangle "link - down" instead of the existing statement. Thus, in this case all the packets sent from site 4 were not delivered through any alternative path until the link is restored.

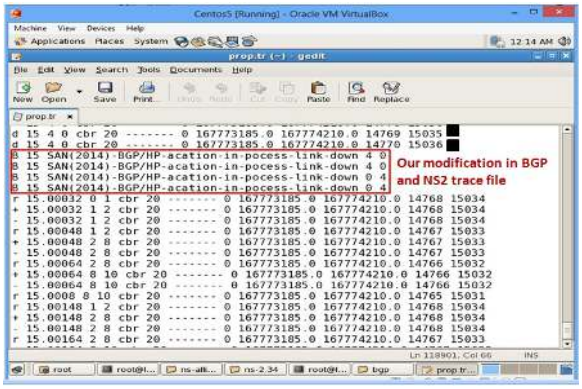

Figure 20: The Modification Of Trace By Using the HP Function

HP is processed for selecting the alternative path by using the closer egress point value from the hash table shown in Figure 20. The success of selecting the alternative path is shown in Figure 20 (NAM) and Figure 22 (trace file).

Figure 21 shows the link between node 4 and node 2 via node 0 fails after starting the simulation. This node 10 in AS1 cannot receive the UDP packets from node 4 in AS0. To keep node 10 still receiving the packets from node 4 with the minimum packet loss as possible, HP selects the alternative path between node 2 to node 4 via node 1 in visualization (NAM).

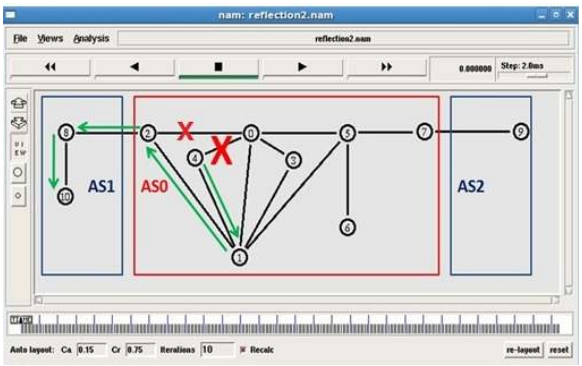

Figure 21: The Alternative Link 4 Between Nodes 4 To Node 2 Via Node 1 Starts To Be Used

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 22 shows HP selecting the alternative path between nodes 2 to node 4via node 1 in the trace file which appears in the black rectangle in the same action as in Figure 20. The alternative path is used until the time reaches 5.5 seconds. At this moment, the link between node 0 and node 2 is restored. Again HP checks the smaller distance to the egress point in the hash table and uses it. Here, there are time gaps between the links down and link up. In the case the link is down, there will be packet loss. However, in the link up case, packet loss is also seen. That is due to HP switching time to the alternative path, while the transmission is already running, casing the packets to be dropped.

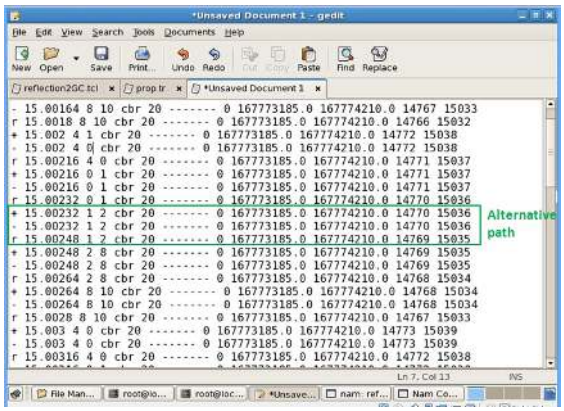

Figure 22: HP Selects The Alternative Path

3.2.7 Result for Different Scenarios

Figure 20 and Figure 22 shows the results between enabling and not enabling HP. In Figure 20, as highlighted in the red rectangle which contains shown the link is down, HP is processing to select the alternative path. In Figure 22, as highlighted in the green rectangle, BGP uses the alternative path.

3.2.7.1 Comparing the Results of ns-BGP with and without HP

The comparison of results before adding HP functionality to ns-BGP and after using the HP in ns-BGP was investigated. Table 8 shows the result of the comparisons for three parameters.

Table 8: The Throughput, Packet Loss And Average Delay Values For Reflection2.tcl Trace File Running HP

Total of packet sends	Total of packets received	Total of packet loss	Average delay Sec	Throughput bit per Sec
79326	79306	20	0.004644769	317970.2

Figure 23, shows the comparison results for three scenarios based on routerelction2.tcl. The comparison parameters include number of packet sent, number of packets received, total packet loss, average delay and the throughput. The first scenario runs the simulation normally without any link failure as mentioned in Figure 23 (based on the original legend). The second scenario runs under two conditions, the first condition shows the link between 0 and 2 down at 1.5 seconds, the second condition shows the same link gets recovered at 5.5 seconds as mentioned in Figure 23shown by the link down legend). The third scenario is similar to second one but during the link down between site 0 and site 2, HP selects the alternative link (site 1 to site 2) as shown in Figure 23 (HP function legend). Each data set in Figure 23 represents the result for the three scenarios parameters stated in Table 3 (Original), with Table 5 (link down) and Table 7 (HP function). The error percentage of packet loss in link down scenario against the original scenario is 22%; the error percentage of average delay (sec) is 12% and the throughput error percentage is 22%. In using HP function scenario, the error percentage of packets loss is 0.03%, the error percentage of average delay (sec) is 0.01% and the throughput error percentage is 0.003%. These comparison shows that HP functionality is very useful in selecting the alternative link with error percentage less than one for all the parameters. Finally, the relation between delay and throughput based on packets send and received was examined. The throughput increases when the delay decreases. The decreases in delay are based on some factors such as packet loss.

3.2.7.2 Validation

The validation is done for the scenarios by using the trace files to calculate the performance parameters. And then comparing the values of throughput, delay and packet loss before and after adding HP functionality to the ns-BGP

Finally, HP algorithm was validated conceptually. The benefits of HP presence are verified. To our knowledge, there is no previous, such practical integration between HP and BGP in network simulators.

The BGP integrated with HP built in this paper through the developed NS-2 script, consists of two parts. These parts are tested and validated. The update messages which should be send

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

repeatedly every 45 seconds by EIGRP has been stopped. The added values are less bandwidth is consumed and as well as less consumption of time and resources for the router. Also, the visualization of the network when using NS-2 is an additional functionality.

3.3 Applying the complete BGP integrated with HP to the Proposed Corporation Network Topology

In the next step, a complete Corporation network using BGP and HP NS-2 model was built. The trace file for baseline validation was also obtained. The trace file inferred from the scenarios will be compared to this trace file.

3.3.1 Dual-Homed network design consisting of 5 sites

The design of dual homed network is explained in the subsection below.

3.3.2 Production network

The original communication network, as shown in Figure 24 (star topology), is an integration of five local sites representing five sites. Each site sends and receives traffic. The sites are connected together through the Network Service Provider (NSP) and these links are used as primary links. The private network links are used as alternative links when there is failure in the primary. STP is used to enable this failover. Each link will have different speeds for NSP but for the private link, the speed is 1250Mbps.

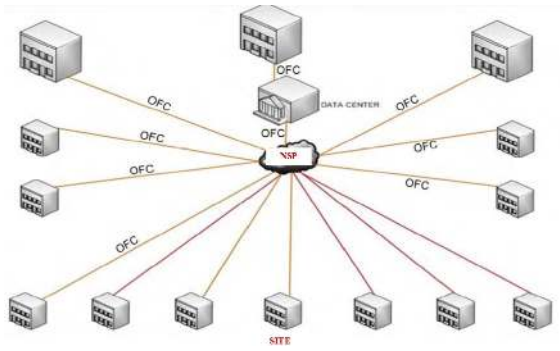

Figure 24: Production network topology

3.3.3 DATA

Traffic data are collected from aggregator router located in Data Center as shown in Figure 25. The router is configured with one port as a mirror port to capture all the traffic using a modified sniffer program coded in C and executed in the computer. The real network traffic parameters are captured at several intervals in the weekend, and saved in a text file. Figure 5.24 shows a snapshot of the resulting captured traffic formatted accordingly after reprocessing.

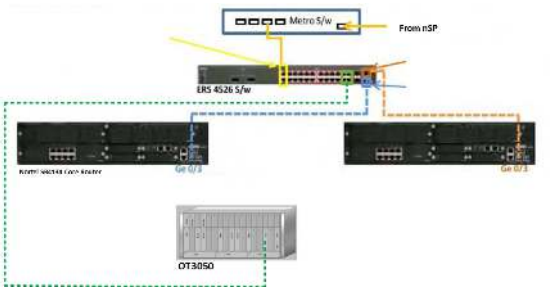

Figure 25:Aggregation router data collection

In Figure 26 the source and destination is represented with network ID without host ID. Also the length of TCP and UDP is small. The reason for that is the processing in the aggregation router is very high during the peak hours. The mirroring the traffic will affect the network immediately due to the processing causing packet drops. Thus the capture was made in weekend.

No	Time	Source	Destination	Pro	Length
1	0.000000	10.2.0.0	10.0.0.0	UDP	56
2	0.143680	10.0.0.0	10.4.0.0	TCP	62
3	0.184793	10.2.0.0	10.0.0.0	UDP	56
4	0.188323	10.0.0.0	10.4.0.0	TCP	62
5	0.221392	10.1.0.0	10.2.0.0	UDP	60
6	0.500999	10.2.0.0	10.0.0.0	UDP	56
7	0.686279	10.2.0.0	10.0.0.0	UDP	56
8	0.723880	10.1.0.0	10.0.0.0	UDP	60
9	1.001866	10.2.0.0	10.0.0.0	UDP	56
10	1.003942	10.0.0.0	10.1.0.0	UDP	60
11	1.186852	10.2.0.0	10.0.0.0	UDP	56
12	1.226414	10.1.0.0	10.2.0.0	UDP	60
13	1.502877	10.2.0.0	10.0.0.0	UDP	56
14	1.688000	10.2.0.0	10.0.0.0	UDP	56
15	1.729035	10.1.0.0	10.2.0.0	UDP	60
16	2.004184	10.2.0.0	10.0.0.0	UDP	56
17	2.189399	10.2.0.0	10.0.0.0	UDP	56
18	2.231771	10.1.0.0	10.2.0.0	UDP	60
19	2.505604	10.2.0.0	10.0.0.0	UDP	56
20	3.006114	10.2.0.0	10.0.0.0	UDP	56
21	3.191659	10.2.0.0	10.0.0.0	UDP	56
22	3.364333	10.3.0.0	10.0.0.0	UDP	56
23	3.507191	10.2.0.0	10.0.0.0	UDP	56
24	3.563883	10.2.0.0	10.0.0.0	UDP	56
25	3.692064	10.2.0.0	10.0.0.0	UDP	56
26	3.861463	10.3.0.0	10.0.0.0	UDP	56
27	4.008219	10.2.0.0	10.0.0.0	UDP	56
28	4.062413	10.2.0.0	10.0.0.0	UDP	56
29	4.362406	10.3.0.0	10.0.0.0	UDP	56
30	4.563075	10.2.0.0	10.0.0.0	UDP	56
31	4.606000	10.0.0.0	10.1.0.0	UDP	60

Figure 26: Example Of Captured Text File

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The proposed network topology shown in Figure 27 is a dual-homed network. It consists of two ASes representing two different states in Sudan which Mirroring port 175 provides AS1 to two links, and connected with AS0. The site 0 is Data center and functions with HP functionality, also site 0 is RR for site 1(RRC) within the cluster ID 1000.

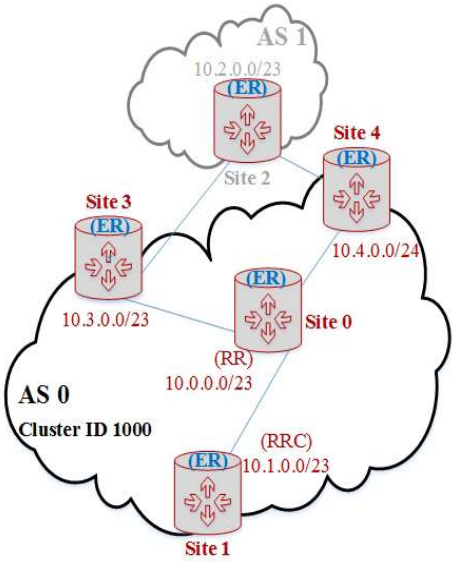

Figure 27: Production Network Topology for 5 Nodes

3.3.4 Maximum number of hops

In this research, the maximum numbers of hops are defined by the longest path in the network topology between two OD pairs and the count of number of hops between them. In this case, the maximum number of hops is three (max_hop =3).

3.3.5 Computing the shortest path

The algorithm assumes that Dijkstra's Minimum Weight Path Algorithm is used to determine the traffic routing. The weights used by Dijkstra's Algorithm are based on Enhanced Interior Gateway Routing Protocol (EIGRP) metric.

The shortest path is calculated based on the given data parameters with the assumptions as below:

Data the parameters of the data are as follows.

1) Cell Size (kb).
2) Link Speed (kbps).
3) Propagation Rate (ms/km).
4) Link Distance (km).
5) Cell Transmission Time ES (ms).
6) Utilization ρ ("from" /"to" direction).

The link information between the sites in Figure 27 are given in Table 9. It includes the direct connection between sites, actual distances in kilometers between them and link speed in megabit per seconds between direct linked sites.

Table 9: The Throughput, Packet Loss And Average Delay Values For Reflection2.tcl Trace File Running HP

Link	Link Speed	Distance
Site 0 __ Site 1	1250Mbps	1 Km
Site 0 __ Site 3	1250Mbps	27 Km
Site 0 __ Site 4	1250Mbps	73 Km
Site 2 __ Site 3	1250Mbps	81 Km
Site 2 __ Site 4	1250Mbps	115 Km

3.3.6 Assumptions

i. Dijkstra's Minimum Weight Path Algorithm is used.
ii. For weights, Enhanced Interior Gateway Routing Protocol
iii. (EIGRP) metric is used.
iv. Using EIGRP's default values for the constant "K" values.

3.3.7 EIGRP cost calculations

The EIGRP weight equation using the default K values is given by equation number (4), and the link delay used in the EIGRP metric equation is number (5).

$$EIGRP = \left[\frac{10^7}{\text{Bandwidth}} + \text{delay}\right] \times 256 \quad (4)$$

$$\text{Link\_delay} = \text{buffering} + \text{transimission\_delay} + \text{propagationdelay} \quad (5)$$

The queuing theory equations are computed for the buffering plus transmission delay based on the type of link traffics (M/M/1) used. The results of all equations give the EIGRP weight for each link endpoint and these results are shown in Table 10.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 10: EIGRP weights for each link in the production network

Link Endpoint	Total packets during period 329 sec	Weight	Packet Avg
(0 --> 1)	230	2000.02	60
(0 --> 3)	89	2000.65	62.13041126
(0 --> 4)	159	2004.57	61.32786885
(1 --> 0)	99	2000.02	60
(2 --> 3)	268	2005.62	56
(2 --> 4)	0	2011.31	62
(3--> 0)	404	2000.65	58.96969697
(3--> 2)	56	2005.62	62.32142857
(4--> 0)	117	2004.57	56.87179487
(4--> 2)	98	2011.31	60

The shortest path for each site is calculated from Table 9. It includes the links endpoints with their weights, for all possible paths between each OD pairs as well as the direct and indirect links.

3.3.8 Simulated the production network

The production network was simulated near to real in NS-2 by using the production network parameters; topology and the captured traffic text file through a modified sniffer C program. The internal NS-2 traffic generators were disabled and packets moving between sites were based on the real network captured traffic during the execution time of simulation TCL file (simulator.tcl). The production network captured data was used in NS-2 instead of their internal traffic generator with 0% error packet send time, 0% packet loss and error received packet time of 3.61%. (Please refer to our paper [13]).

Finally, the developed ns-BGP was integrated with HP as an added feature. This feature provides all the links weight for possible scenarios in network topology, such as link down to update the HP automatically.

3.3.8.1 Evaluation for HP enabled scenarios

The performance evaluation of each scenario is based on three parameters i.e. total packet loss, average delay and throughput. Firstly, the values of the parameters for the dual-homed network topology must be obtained. Secondly, the scenario for link failure will be. Finally, the first result which forms as a baseline will be compared with the second result.

3.3.8.2 Dual-Homed Network Topology

The trace file for this topology is used to compare the result with different scenario. Figure 28 shows the design run in NS-2. The previous equations (1, 2 and 3) were applied to get the total packet loss, average delay and throughput values. The results are given in Table 11.

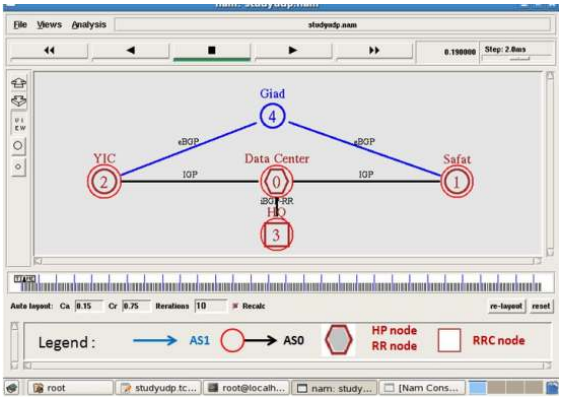

Figure 28: The dual-homed network for corporation

Table 11: The Throughput, Packet loss and average Delay values based on reflection2.tcl trace file

Total of packet send	Total of packet received	Total packet loss	Average delay sec	Throughput bit per sec
1000	1000	0	0.08796583	1415.127273

3.3.8.3 Link Failure Scenario

As shown in Figure 29, the link between site 0 and site 2 is down at 1.5 second and restored at 5.5 seconds (simulation time). The ns-BGP integrated with HP was thereafter evaluated in handling the failure. Here, the HP processing time to select the alternative path is 0.05 seconds.

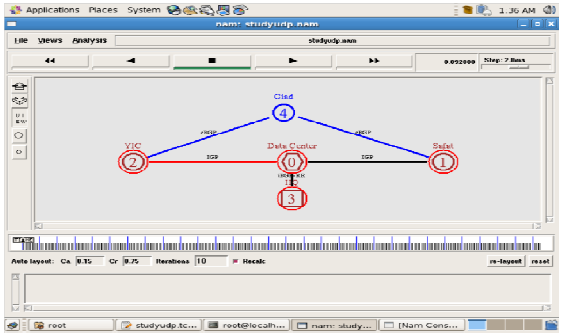

Figure 29: The link down and up scenario between site 0 and site 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

NS-2 model (BGP integrated with HP) moves all the traffic between site 0 and site 3 to the link which connects the site 0 to site 3. This links has smaller egress point distance to AS 1. After the link recovers at 5 seconds, HP recalculates to select the smaller egress point distance. Here, HP selects the older link. The three parameters and their values are given in Table 12.

Table 12 : The Throughput, Packet loss and average Delay values for reflection2.tcl trace file running HP

Total of packet send	Total of packet received	Total packet loss	Average delay sec	Throughput bit per sec
1000	998	2	0.088152866	1412.266667

Table 11 shows that the packet loss is only 2 packets. This loss appears because HP searching time in the hash table is 0.05 seconds. So, if a dedicated CPU such as router is used, then the CPU time will decrease.

3.3.8.4 Validation HP Scenario (Link Failure Scenario)

The results between the baseline topology and link failure topology were compared. The comparison between the ns-BGP integrated with HP trace file and the scenario is again based on comparing the values for the three metrics. The results show that ns-BGP is working with HP functionality. HP learns the routing update from the hash table a delay equals to 0.05 seconds. The value for the simulation again depends on the computer specification. Figure 30 shows the comparison between ns-BGP without and with HP functionality.

3.3.9 Real network scenario consists of 32 nodes

Figure 31 illustrates the network topology consisting of 32 sites; there are three ASes, AS0, AS1 and AS2. These ASes represents three different geographical locations. In AS0 there is one HP site (site 0) running as RR and there are 17 RRC sites and non RRC 4 sites used for connection between AS1 (site 3 and site 4) and AS2 (site 1 and site 5). In AS1 there is one HP site (site 23) which is running as RR and there are two sites (site 2 and site 22) used to connect to AS0. In AS2, there is one HP site (site 27) which is running as RR and there are 4 RRC sites and two non RRC client (site 25 and site 28) used to connect with AS0.

For the three HP sites involving AS0, AS1 and AS2, loading of the hash table takes 0.05 seconds. The three metrics used in evaluating the performance of HP functions are the packet loss the packets loss error percentage with a value of 0.03%, the error percentage of average delay (Sec) of 0.01% and the throughput error percentage of 0.003%. These values are calculated by using AWK.

4. CONCLUSION

The Integrated Model is introduced consisting of HP algorithm and BGP integrated into the NS-2 network simulator has been built. The information handling from hash table for HP is seamless with HP routing table dynamically updated automatically to keep the BGP functionality healthy on delivering the packets to the destination without any loss or drop packets. Visualization is provided into NS-2. Ns-BGP with HP is evaluated by running different scenarios, such as link failure. The trace file for each scenario is compared with the baseline ns-BGP standard trace file based on the original network topology for benchmarking and validation. The maximum numbers of sites in the network topology evaluated by model are 32 nodes and were divided into three ASes. Each AS has site acting with HP functionality.

The overall convergence of the BGP and HP model for NS-2, the analysis of scenarios shows that HP needs 0.05 seconds to switch from the primary link to the alternative link. The three metrics, packet loss, average delay and throughput, are used in benchmarking the trace file result against the real production network captured text file. The result shows packets loss error percentage of 0.025%, the error percentage of average delay (Sec) is 0.013% and the throughput error percentage is 0.003.

5. FUTURE WORK

The research is not exhaustive in proposing a fully Integrated Simulation Model (ISM) based on the components mentioned earlier. The load balancing is yet to be incorporated into the ISM where all links are fully utilized at all-time rather than just having passive links which are dormant. Furthermore, this justifies the need for more investment in laying more cables in the network. Thus, this research can be extended

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

toward achieving better load balancing network. Also, ISM implements the intent of the production network. The operating systems of the routers have to be modified for this purpose. The following further describes this direction.

REFRENCES:

[1] J. Pan and R. Jain, "A survey of network simulation tools: Current status and future developments," *Email: jp10@ cse. wustl. edu,* 2008.

[2] A. M. Gosai and B. H. Goswami, "Network Simulator for Efficient Performance Parameter Testing & Evaluation," *National Journal of System and Information Technology,* vol. 5, pp. 89-105, 2012.

[3] T. Issariyakul and E. Hossain," Introduction to network simulator NS2:" *Springer*, 2011.

[4] I. S. Ahmad, A. Kalakech, and S. Kadry, "Modified Binary Exponential Backoff Algorithm to Minimize Mobiles Communication Time," 2014.

[5] S. Kurkowski, T. Camp, and M. Colagrosso, "MANET simulation studies: the incredibles," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, pp. 50-61, 2005.

[6] "The Network Simulator - NS-2 " http://www.isi.edu/nsnam/ns/ .

[7] J. K. Ousterhout, "TCL and the TK Toolkit," *Computer Science Division, Department of Electrical Engineering and Computer Science, University of California, Berkeley,* 1993.

[8] A. Robbins and N. H. Beebe, "A Bibliography of Classic Shell Scripting," 2011.

[9] S. Rampfl, "Network Simulation and its Limitations," *in Proceeding zum Seminar Future Internet (FI), Innovative Internet Technologien und Mobilkommunikation (IITM) und Autonomous Communication Networks (ACN)*, 2013.

[10] B. Premore, " SSFNet BGP User's Guide: http://www.ssfnet.org/bgp/user-guideps.," 2004.

[11] SSFNet,"SSFNet."http://www.ssfnet.org/," 2003.

[12] T. D. Feng, "Implementation of BGP in a network simulator," Citeseer, 2004.

[13] S. A. NAGAR, S. M. NOR, and M. S. BOBA, "GENERATION OF REAL WORLD TRAFFIC USING NS2 TRAFFIC AGENTS," *Journal of Theoretical and Applied Information Technology,* vol. 67, 2014.

[14] R. Pethe and S. Burnase, "Technical Era Language of the Networking-EIGRP," *International Journal of Engineering Science and Technology, NCICT Special Issue,* pp. 1-5, 2011.

[15] A. U. Salleh, Z. Ishak, N. M. Din, and M. Z. Jamaludin, "Trace analyzer for ns-2," in *Research and Development, 2006. SCOReD 2006. 4th Student Conference on*, 2006, pp. 29-32.

[16] R. Teixeira, S. Agarwal, and J. Rexford, "BGP routing changes: merging views from two ISPs," *SIGCOMM Comput. Commun. Rev.,* vol. 35, pp. 79-82, 2005.

[17] P. Jong Han, R. Oliveira, S. Amante, D. McPherson, and Z. Lixia, "BGP route reflection revinoded," *Communications Magazine, IEEE,* vol. 50, pp. 70-75, 2012.

[18] B. Quoitin and S. Uhlig, "Modeling the routing of an autonomous system with C-BGP," *Network, IEEE,* vol. 19, pp. 12-19, 2005.

[19] R. Teixeira, A. Shaikh, T. Griffin, and G. M. Voelker, "Network sensitivity to hot-potato disruptions," *SIGCOMM Comput. Commun. Rev.,* vol. 34, pp. 231-244, 2004.

[20] B. H. a. D. M. Pherson, " Internet Routing Architectures,," *Cisco Press,* vol. 2nd ed, 2000.

[21] R. Teixeira, A. Shaikh, T. G. Griffin, and J. Rexford, "Impact of hot-potato routing changes in IP networks," *IEEE/ACM Trans. Netw.,* vol. 16, pp. 1295-1307, 2008.

[22] A. B. Hassouna, H. Koubaa, and F. Kamoun, "A model for deploying an opportunistic MAC protocol in NS-2," in *2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, 2012, pp. 604-611.

[23] Caesar, M. & Rexford, J. "BGP routing policies in ISP networks" *Network,IEEE, 19*, 5-11.2005.

[24] OPRESCU, M. I. "Virtualization and distribution of the BGP control plane".2012

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[25] N. Bhargava, R. Bhargava, B. Kumar, S. Gupta, N. K. Senwaliya, and K. K. Jyotiyana, "Implementation of TCP_Reno Algorithm in the ns2." *International Journal of Engineering and Innovative Technology (IJEIT)* Volume 2, Issue 2, August 2012.

[26] M. Kumar and G. Kumar, "To Analyze and Compare Ring and Mesh Topologies with Varying Traffic Patterns," *parameters,* vol. 1, 2016.

[27] Y. Gheraibia and A. Bourouis, "Ontology and automatic code generation on modeling and simulation," in *2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, 2012, pp. 69-73.

[28] Griffin, T., Resende, M., Rexford, J. & Teixeira, R. "Traffic engineering method with tunable inter-domain egress selection". *U.S. Patent No. 7,904,586*, issued March 8, 2011.

[29] Gutierrez, P. A. A., Pöyhönen, P., Gamir, L. E. I. & Ferrer, F. H. " Using BGP-4 to Migrate to a Future Internet. Mobile Networks and Management". *Springer.*2011.

[30] Vissicchio, S., Cittadini, L. & Di Battista, G."On iBGP Routing Policies. Networking," *IEEE/ACM Transactions on, 23, 227-240.*2015.

[31] Sahraei, M. R. (2009). Integration of ns-BGP with NS-2.34.

[32] Teixeira, R., Shaikh, A., Griffin, T. & Rexford, J. "Dynamics of hot-potato routing in IP networks." *ACM SIGMETRICS Performance Evaluation Review,*32, 307-319,2004.

[33] Teixeira, R. "Network sensitivity to intradomain routing changes", 2005.

[34] Li, K., Wang, S., Xu, S., Wang, X., Ren, J., Huang, H. & Zhai, B. "Multi-topology routing based egress selection approach to achieve hybrid Intra-AS and inter-AS traffic engineering". *International Journal of Communication Systems, 28, 1551-1571,*2015.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

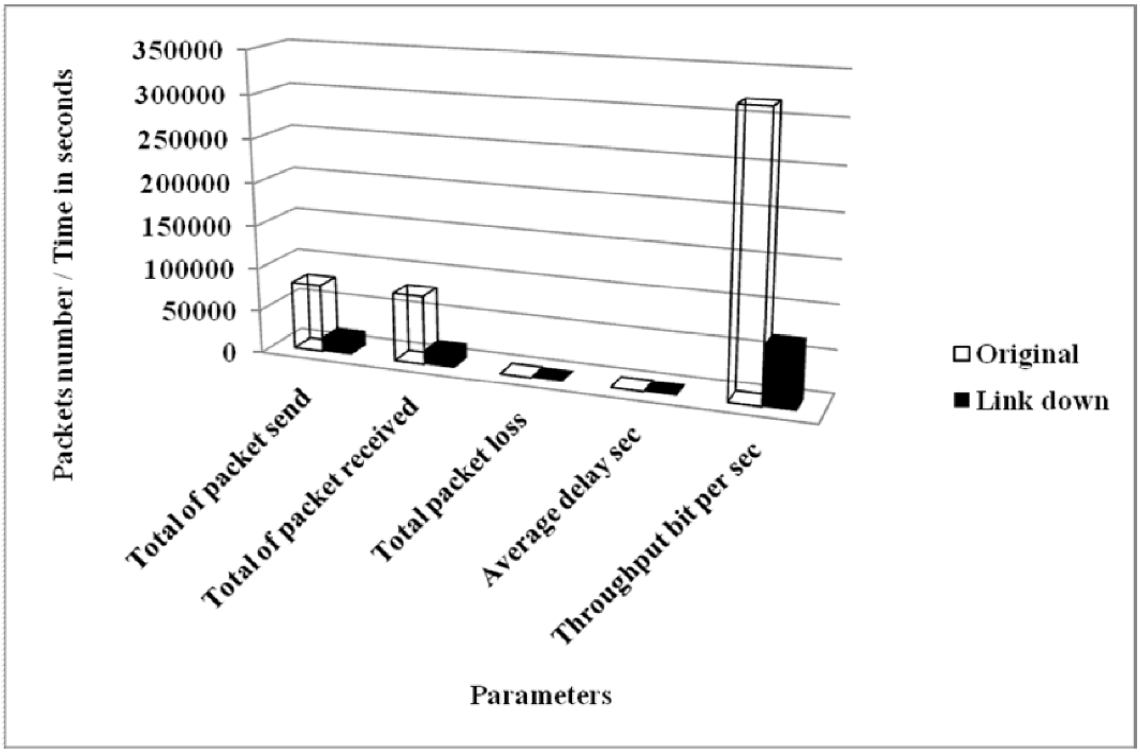

Figure 14: Comparisons between the trace file without link failure and with link failure

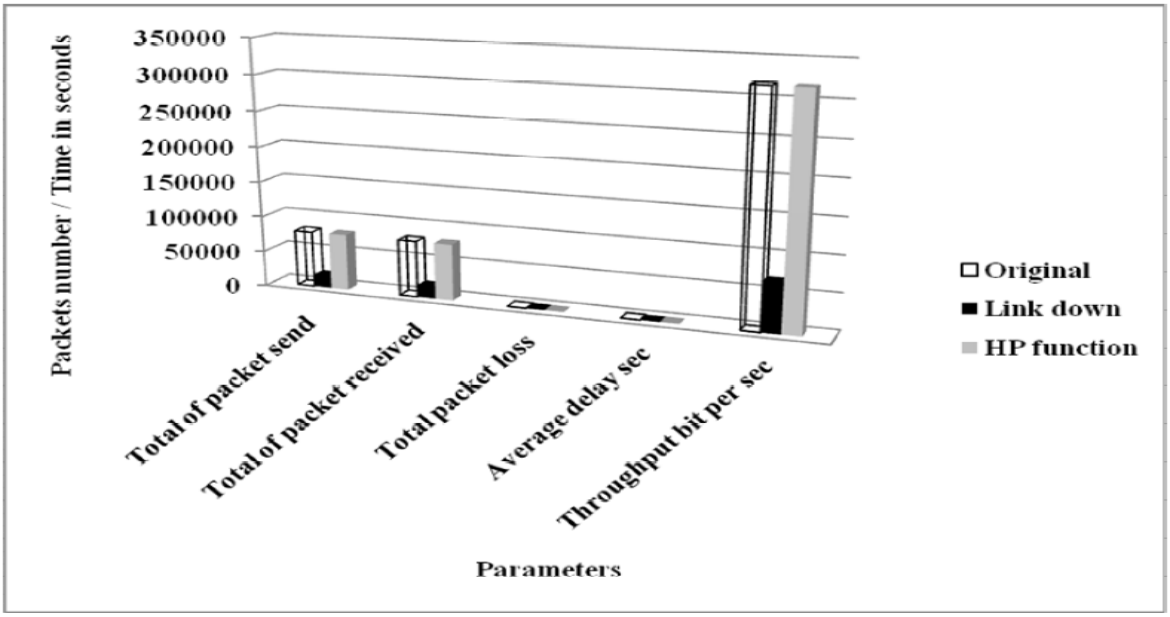

Figure 23: Comparisons Between The Trace Files For Both Without Link Failure and With Link Failure

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

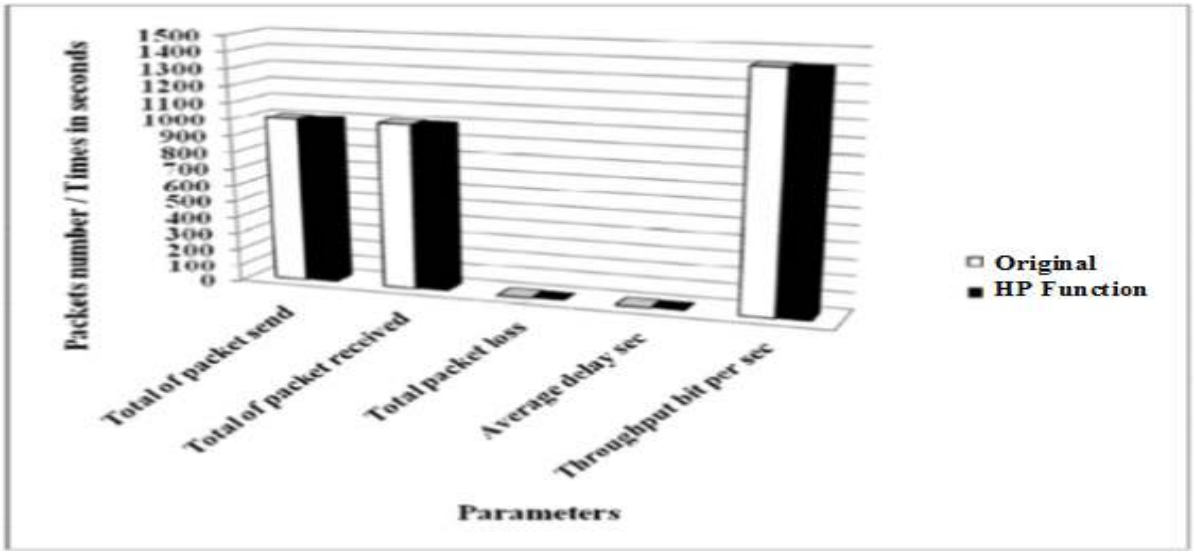

Figure 30: Comparison between HP and link failure scenarios based on Throughput, Delay and Packets loss

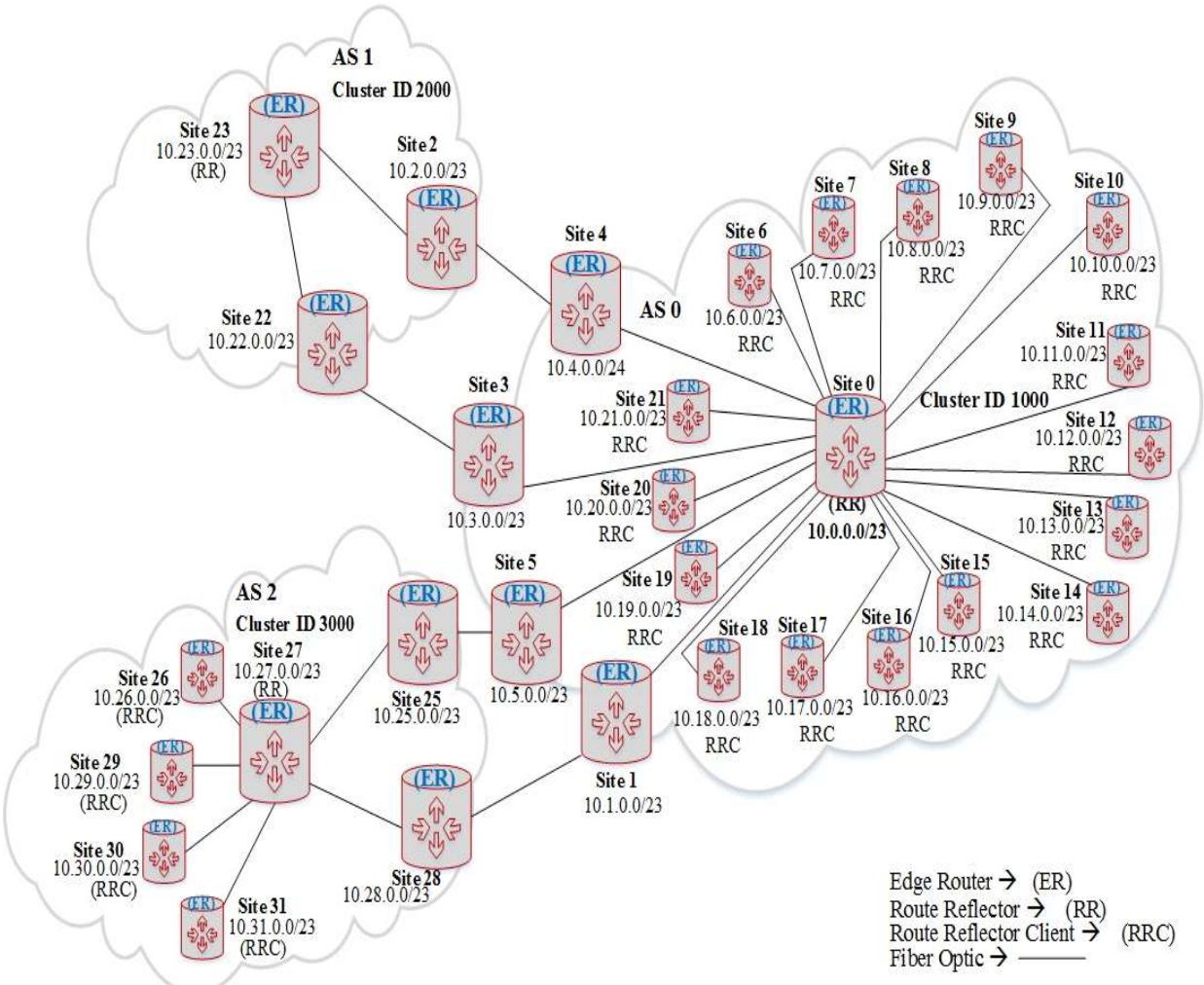

FIGURE 31: PRODUCTION NETWORK. PRODUCTION NETWORK TOPOLOGY FOR 32 NODE