NETWORK MANAGEMENT CONTROL BASED ON CHANGE OF SELF – SIMILARITY AND FUZZY LOGIC IN NETWORK TRAFFIC

Akhigbe – Mudu Thursday Ehis, F. T. Ibharalu

Department of Computer Science, Federal University of Agriculture Abeokuta. Nigeria.

ABSTRACT: Management issues in a network communications are complex, difficult and the size of the networks make network management a daunting task. It is a broad term that includes numerous complex mechanisms for optimizing network performance. Hence, a suitable and effective approach is needed in real time for resource utilization. This paper describes a novel idea of using Fuzzy Logic (FL) for effective network management. FL statistics is based on the likelihood function of the model which is applied to detect point of traffic failure by comparing the null hypothesis against the alternative hypothesis. In our approach, the structure of Fuzzy decision has two dimensions: Input/output. The inputs are the Hurst parameters and its changing rate, while the output is the influence of traffic intensity on network performance. We employed Entropy Measure to validate the functionality of our approach and the result demonstrates that the approach is sufficient for traffic classifications.

KEYWORDS: networks, management, self – similarity, Fuzzy Logic, degradation

1. INTRODUCTION

Management issues in a network are complex and difficult to accomplish. In spite of the initial configuration and installation of network devices, networks need continuous maintenance for optimal utilization with the help of monitoring and management tools. Network management is a broad term that includes numerous complex mechanisms for optimizing network performance, such as congestion control, resource allocation, loads balancing, workload characteristics etc. Dynamic real – time management can improve network utilization and limit the down time of network devices, which can result in improving the user perceived network performance. In a large and complex networks, there are multiple points of failure, which might includes the input buffer at the intermediate router getting filled up and packets are dropping as a result, or the proxy server might be overloaded hence requests are dropping, or some WAN Link might itself be down etc. Continuous measurement and monitoring of network states such as bandwidth, link utilization, packet drop rate and link delay, help in detecting and diagnosing these failures. But size and complexity of network make measurements challenging task. Measurement of network though challenging, provides valuable information about improving performance, assessing utilization, engineering traffic and validating design choice [SNB04].

Thus, effective and real- time management of networks is vital for optimal resource utilization and network performance. The complex areas of network management can benefit a lot when integrated with machine learning (ML). By complex areas we mean areas like resource allocation in QoS (Quality of Service) aware architecture, policy branch of machine learning, namely Fuzzy Logic (FL), to simplify complicated areas of network management [WY08]. Fuzzy Logic has a great resemblance with human decision making ability, as it generates precise solutions and results from certain approximate reasoning technique that takes imprecise data and produces intelligent results. Fuzzy Logic can take data and produce intelligent solutions for management purposes [RMS08].

1.1. Statement of the Problem

This paper has four major contributions: (1) it proposes network traffic management using Fuzzy statistics to determine the change point of self similarity. (2) It monitors the network performance in terms of failure and we can detect and diagnose the exact cause of failure. (3) it develops a tool which can keep track of network status, and help network users to diagnose problems effectively and efficiently. That is, it correctly and quickly detects, and diagnoses the cause of failure. The rest of this paper is structured and organized as follows: Section 2, discusses some related work. Section 3: discusses the concepts of fuzzy logic. Section 4: discusses different features of the network management that makes fuzzy logic suitable to be used in this area. Section 5: briefly describes the methodology. Section 6: discusses the implementation and evaluation and section 7 concludes the paper.

2. LITERATURE SURVEY

In this section, we summarize the work done by others in the field of network management issues. It is not a comprehensive summary but summary of literature survey done by others to measure and

Tibiscus

manage network properties. Internet is one of the fastest growing means of information flow today and it serves a wide range of people from an average web browser to system administrators. However, when in operation, packet level measurement is critical to ensure the desired performance metrics are achievable [BST12]. Information sharing over computer network requires a reliable, secure path for data delivery, hence, [BCB12] presents a technique for selecting the most reliable path for communication between node pairs of a computer network. Reliability has been calculated based on the bandwidth utilization by the nodes. [DRS12] stressed that the field of evaluation of user interface known as usability evaluation has become increasingly important with computer users. Usability, they further stressed is the ease with which a user can learn to operate, prepare inputs for and interpret outputs of a system or components. Usability also became a key issue in Human Computer Interface (HCI) because not only it provides descriptive background and guidelines for development of a quality user interface but it is also concerned with supporting users during their interactions with computers [DRS12]. The research done by [DRS12], mathematically proved that there is a statistically significant change in the average Hurst parameter network degradation occurrence. [GB12] proposed a method using Hurst parameter to identify network degradation, which causes a decrease in the traffic's self similarity. This method consider the normal range of Hurst parameter to be [0.5 – 0.9] and there is network failure when the Hurst parameter runs out of this range. Nevertheless, all of these existing methods can only detect the presence of attack / failure, they cannot identify at what time the network failure happened. Fuzzy Logic is one of the most popular methods used in attack detection for it can deal with the vague and imprecise boundaries between normal traffic and different levels of attacks [ZSJ10; BCB12]. [OCI13] propose to use the fuzzy logic to analyze the Hurst parameter and estimate the time duration of network degradation. The major contribution of [ZSJ10] method is considering the inherent relationship between Discrete Wavelet Transform (DWT) and self similarity, and propose to use Schwartz Information Criterion (SIC) combined with DWT to detect the occurrence of DDoS (distributed denial of Service) flood attack, therefore real time DDoS attack detection is achieved through this method. Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth values between "completely true" and "completely false" [OCI13]. They suggested that logic underlying modes of reasoning are approximate rather than exact. The importance of fuzzy logic derives from the fact that most modes of human reasoning and especially common sense reasoning are approximate in nature. [OCI13] further stressed that fuzzy logic is a problem solving control system methodology, capable of generating conclusions based upon vague, ambiguous imprecise input information. Fuzzy Logic is currently preferred in control systems because of it's' robustness and does not insist on noise free inputs and can implement non linear systems without any known mathematical models. The output control is usually a smooth control function even when a wide range of input variations exist [RV12].

2.1. A Brief Review of Self- Similarity

Self similarity means that the sample paths of the process $\omega(t)$ and those of rescaled version $C^H\omega(t/c)$ obtained by simultaneously dilating the time axis t by a factor $c \geq 0$, and the amplitude axis by a factor C^H, cannot be statistically distinguished from each other [ZSJ10]. Equivalently, it implies that an affine dilated subset of one sample path cannot be distinguished from its whole. H is called the self – similarity or Hurst parameter. For a general self – similarity process, the parameter H measures the degree of self- similarity [SSK12].

2.2. Fuzzy Logic System

Fuzzy Logic has a great resemblance with human decision making ability, as it generates precise Solutions and results from certain approximate reasoning technique that takes imprecise data and produces intelligent results. Fuzzy Logic can take data and produce intelligent solutions for management purposes [RMS08]. Fuzzy Logic disposes information based on fuzzy reasoning rules. It makes self adaptive decision in light of mature experience. Fuzzy Logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth values between "completely true" and "completely false" [OCI13]. As its name suggests, it is the logic underlying modes of reasoning which are approximate rather than exact. The importance of fuzzy logic derives from the fact that most modes of human reasoning and especially common sense reasoning are approximate in nature.

3. METHODOLOGY

3. 1. Change of Point Estimation

The traffic inputs are live streams from measurement of live links. To achieve our goal, a frame work algorithms called Traffic monitor algorithms is designed (see figure 1 and figure 2) respectively, to carry out a real time network management analysis. Traffic is captured through a sample run of this algorithm displayed in figure 3 below. To manage the capturing and sampling, two processes are used: one to capture the traffic on a per packet basis and update the appropriate packet counters, and the other to access these counters every second. The output window is then processed, for purpose of normalization and. The window we work with is five minutes long, i.e. five minutes worth of traffic (these values are consistent with general network monitoring practices) [AIF12].

Suppose a sequence of length Q is given, and suppose there is only one change point at position $h.(1\angle h\angle Q)$. Our approach has the merit of detecting the change point in the variance structure of the sequence. This is accomplished by computing the variance of the entire sequence and the pairs of pieces $\left(f_1 = 1,...h.and.f_2 = h+1,...,Q\right)$, compare their values and then decide if there is a change point at position h, by testing the null hypothesis, H_0 (no change is present) against the alternative H_1(a single change is present). The statistics for the two hypothesis is given by

$$H_0 : stat(Q) = \sqrt{\frac{(n_1-1)\sigma_1^{\,2} + (n_2-1)\sigma^2_{\,2}}{n_1+n_2-2}} \quad \text{and}$$

$$H_1 = stat(h) = \sqrt{\frac{(n_2-1)\sigma_2^{\,2} + (n_3-1)\sigma_3^{\,2}}{n_2+n_3-2}}$$

where $\sigma^2, \sigma_1^{\,2}, \sigma_2^{\,2}.and.\sigma_3^{\,2}$ are the unbiased maximum likelihood estimators (MLE) of the variances of the entire sequence and of the first and second pieces respectively? Our decision follows the calculated information, that is, H_0 will not be rejected if $stat(Q) \le stat(h)$, otherwise, H_0 will be rejected if: $stat(Q) \ge stat(h)$. The change point at position h can be estimated according to:

$$stat(h) = \min_{1\angle h\angle Q} stat(h) \qquad (1)$$

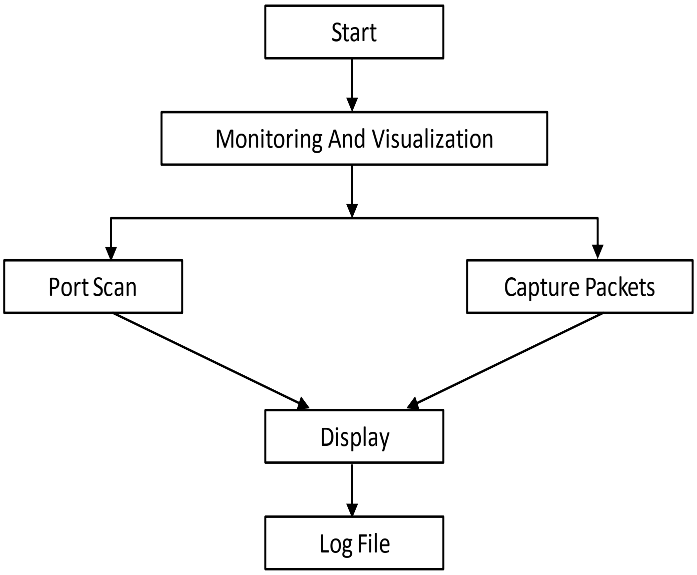
3.2. Traffic Monitor Algorithms

3.2.1. Traffic Monitor Algorithms

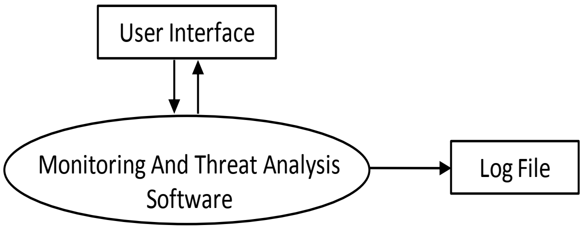
```
(A)      Algorithm for Port Scanning
Step 1:  Variable Declaration
Declare variables for storing IP Address and host name
and set them to null
Step 2:  Input
         2.1 Enter value of Host name (or IP Address)
Step3:   Scanning
         3.1 Declare variable port = 0
         3.2 Declare initial port = value.
         3.3 Declare final port = value.
         3.4 Check if the port is available between
initial port and final port.
         3.5 increment port by 1
         3.6 Repeat step 3.4 up to final port.
Step4:   Display
         4.1 Display all the active ports in GUI format
(B)      Algorithm for packet capturing
Step 1:  Obtaining the list of network interfaces
         1.1 Create a variable array of devices
         1.2 Detect network interfaces present in user
         1.3 Store the above list in devices variable.
Step 2:  Displaying the list of network interfaces
Declare loop counter integer variable i and initialize to
0
While the value of i is less than the length of the array
of devices, do Step 2.3
Print out the name and description of the captured
Network Interface.
Step 3:  Open the network interface.
Declare integer variable J and initialize to  zero (J=0)
While J < length of array of devices, Goto Step 3.3 else
Goto Step 3.7
Check if the network interface at Jth index number in
devices array is selected. If yes goto Step 3.6 else goto
Step 3.4
J=J+1 Goto Step 3.2
Open the selected network interface i.e. Network
Interface at Jth index, then Goto Step 4
Display that the network interface has not yet been
selected by the user. he user. Goto Step 8
Step 4:  Capture packets from the network interface
Is the menu button of stop capture packet selected? If
yes goto Step 3.8 else goto Step 4.2
Capture the upcoming single packet from the network
Display the captured packet by going to Step 5
Step 5:  Display the captured packet to the user in
proper GUI format.
Detect user' menu choice of the format in which
captured packet's to be displayed
Analyze the packet. Display in Hexadecimal format
Goto Step 6 to save the packets to a temporary file
Go back to Step 4.1
Step 6:  Save captured packets into a file
         6.1 Create a temporary file say
         6.2 Save captured packets into the opened file
         6.3 Go back to Step 5.4
Step 7:  Close all the open network interface
         7.1 Delete the temporary file.
7.2 Close the network interface.
Step 8:  End
```

Figure 1: Traffic monitor algorithm [AIF12].

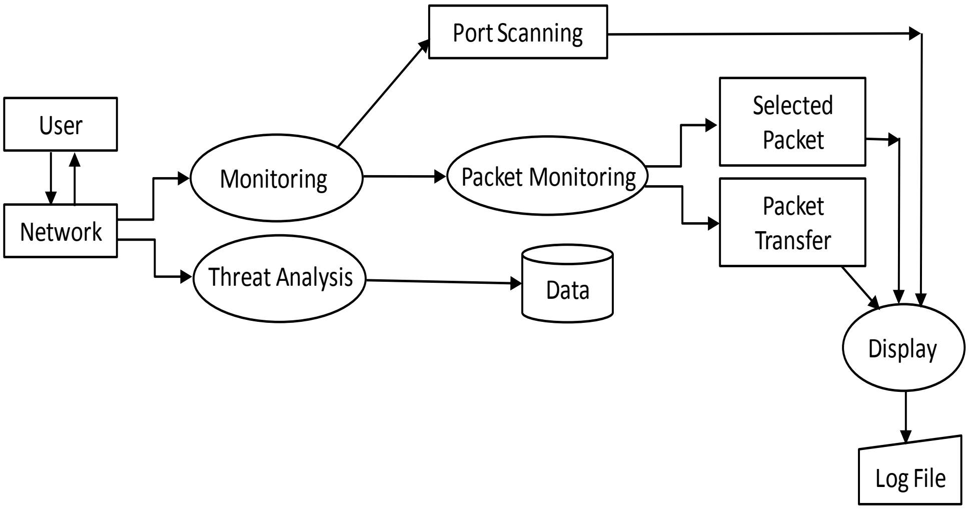
Tibiscus



Flow chart for monitoring And visualization

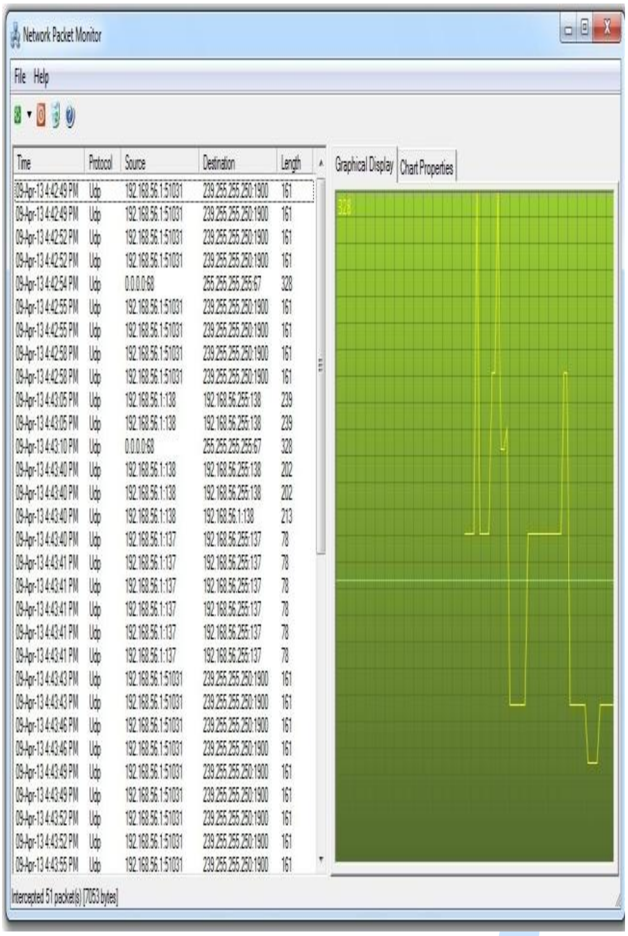


Context Diagram



Data Flow Diagram

Figure 2: Flow Charts for Packets Monitor



Time	Protocol	Source	Destination	Length
09-Apr-13 4:42:49 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:49 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:52 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:52 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:54 PM	Udp	0.0.0.0:68	255.255.255.255:67	328
09-Apr-13 4:42:55 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:55 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:58 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:42:58 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:05 PM	Udp	192.168.56.1:138	192.168.56.255:138	239
09-Apr-13 4:43:05 PM	Udp	192.168.56.1:138	192.168.56.255:138	239
09-Apr-13 4:43:10 PM	Udp	0.0.0.0:68	255.255.255.255:67	328
09-Apr-13 4:43:40 PM	Udp	192.168.56.1:138	192.168.56.255:138	202
09-Apr-13 4:43:40 PM	Udp	192.168.56.1:138	192.168.56.255:138	202
09-Apr-13 4:43:40 PM	Udp	192.168.56.1:138	192.168.56.1:138	213
09-Apr-13 4:43:40 PM	Udp	192.168.56.1:137	192.168.56.255:137	78
09-Apr-13 4:43:41 PM	Udp	192.168.56.1:137	192.168.56.255:137	78
09-Apr-13 4:43:41 PM	Udp	192.168.56.1:137	192.168.56.255:137	78
09-Apr-13 4:43:41 PM	Udp	192.168.56.1:137	192.168.56.255:137	78
09-Apr-13 4:43:41 PM	Udp	192.168.56.1:137	192.168.56.255:137	78
09-Apr-13 4:43:41 PM	Udp	192.168.56.1:137	192.168.56.255:137	78
09-Apr-13 4:43:43 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:43 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:46 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:46 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:49 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:49 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:52 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:52 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161
09-Apr-13 4:43:55 PM	Udp	192.168.56.1:51031	239.255.255.250:1900	161

Intercepted 51 packet(s) [7053 bytes]

Figure 3: Output Window of the Algorithms

3.3. Network Management Analysis

The purpose of sample survey design is to maximize the amount of information for a given cost.

Population means, $\mu = \dfrac{\sum_{i=1}^{n} \chi_i}{N}$ (2)

Sample mean, $\overline{\chi} = \dfrac{\sum_{i=1}^{n} \chi_i}{n}$ (3)

where n is the sample size.

Population variance, $\sigma_X^2 = \dfrac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{N-1}$ (4)

And

Sample variance, $S^2 = \dfrac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}$ (5)

Solving these equations in practical situations presents a problem because the population variance, σ_X^2, is unknown, but S^2, is available from prior experimentation and can therefore, replace σ_X^2 with S^2, in equation (4). $n(Q)=30$ and segment the sequence into three pieces of 10 each i.e. $n(h_1)=10, n(h_2)=10, n(h_3)=10$ respectively. Q represents the entire sequence.

From figure 3: (output window of the algorithms) -

$$h_1 = \{161,161,161,161,328,161,161,161,161,239\}$$

$$h_2 = \{239,328,202,202,213,78,78,78,78,78\}$$

$$h_3 = \{78,161,161,161,161,161,161,161,161,161\}$$

Mean of $\bar{h}_1 = 185, \bar{h}_2 = 157, \bar{h}_3 = 153$ and $\bar{Q} = 165$ computed using equation (3)

$$\sigma_1^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1} = 3{,}108,$$

$$\sigma_2^2 = 8{,}261, \sigma_3^2 = 689$$

$$H_0 : stat(Q) = \sqrt{\frac{(n_1-1)\sigma_1^2 + (n_2-1)\sigma_2^2}{n_1+n_2-2}} = 75.4$$

$$H_1 : stat(h) = \sqrt{\frac{(n_2-1)\sigma_2^2 + (n_3-1)\sigma_3^2}{n_2+n_3-2}} = 66.9$$

Decision: H_1 is accepted which implies that a change point exists and the change point can be estimated using equation (1). Stat (Q) symbolizes the pooled estimate for the standard deviation and pooled in this case means that the information from both samples is combined so as to give the best possible estimate [Joh76].

3.4. Fuzzy Decision Rules on Traffic Classifications.

In this paper, we propose traffic classification decision system using Fuzzy mathematics. The traffic classification we refer here includes light traffic intensity, if the network experiences a slight decline in performance, moderate when the traffic load is high but considered as severe if it experiences serious decline in performance [ZSJ10]. Fuzzy Logic offers a convenient way to produce a mapping between input and output spaces but using natural expressions. It affords the opportunity to approach a fuzzy Logic models without data [DRS12]. To evaluate a Fuzzy usability, three main Linguistic variables are

Tibiscus

defined:- (LA, MA, SA) where $"LA","MA",and."SA"$ represents Light Traffic Intensity, Moderate and Severe Traffic Intensity respectively. [ZSJ10] method is used for defining Fuzzy Rule due to its simplicity. In our approach, the structure of Fuzzy Decision is two dimensional: input/output. The two inputs are the Hurst parameter and its changing rate. The Hurst parameter refers to the dynamic traffic intensity and the changing rate refers to the influence of this traffic intensity on network performance. The output is the influence of traffic intensity on performance. As exhibited in figure 4, the fuzzy decision process of the traffic intensity consists of three parts: Hurst Parameters and its changing rate. When the Hurst parameter is considered moderate, we infer there is a light traffic if the changing rate of the Hurst parameter is considered small. In a similar way, there is moderate traffic intensity if the changing rate of the Hurst parameter is moderate and severe traffic intensity if the changing rate of the Hurst parameter is big (see figure 4).

HC'	*H'*		
	S	M	B
S	MA	LA	LA
M	SA	MA	LA
B	SA	SA	MA

Figure 4: Fuzzy Decision Rules

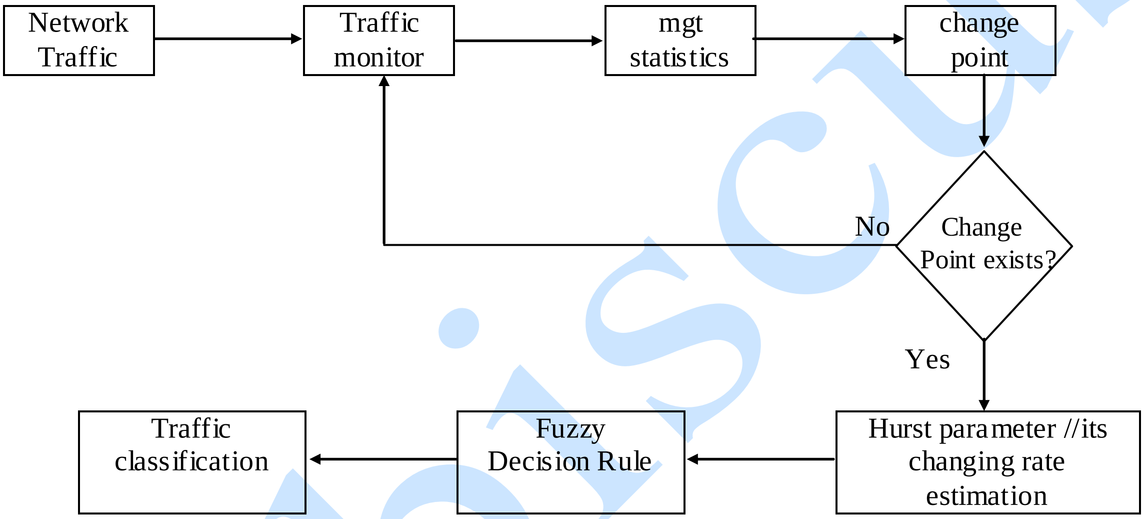


Figure 5: Traffic Classification

3.5. Performance Evaluation

Entropy is a measure of disorder of a network system. If the system tends to be in disorder, its entropy increases towards 1; if the system tends to be in order, then its entropy decreases towards 0. We can view certain attributes of packets that we capture in a period of time as a set. The entropy of the packet attribute – value can be defined as:

$$H(P) = -\sum_{i=1}^{n} p(\chi_i)\log(\chi_i) \qquad (6)$$

where χ_i is the sum of packets with certain attributes, and with a minus sign to get a positive quantity? This approach can be used to check for the differences between the normal and abnormal network action [ARA12]. Computing certain attributes existing in the sequence; $\{g_1 = 8, g_2 = 9, g_3 = 5\}$ and summing up to 22 attributes. Therefore applying equation (6): gives 0.09 which is tending to zero. This assumption justifies the network in normal state (figure 5). We determined variables that characterize network behavior and developed models for network traffic classification (see figure 6).

4. CONCLUSION AND FUTURE WORK

This paper describes a novel idea of using Fuzzy Logic, for effective network management based on change of self similarity in a network. The fuzzy statistics is based on the likelihood function for the model, and is applied to the classification of traffic by comparing the likelihood of the null hypothesis against the alternative hypothesis. We design a frame work algorithms called traffic monitor algorithms, to carry out a real time network management analysis. Fuzzy offers a convenient way to produce a mapping between input and output spaces by using natural expressions. In our approach, the structure of fuzzy decision is two dimensions: Input and Output. The inputs are the Hurst parameters and its changing rate

Tibiscus

while the output is the influence of traffic intensity on network performance. For a functional traffic classification, we employed Entropy Stability Measures to test the validity of our approach and the result demonstrates that the approach can effectively and intelligently classify traffic intensity.

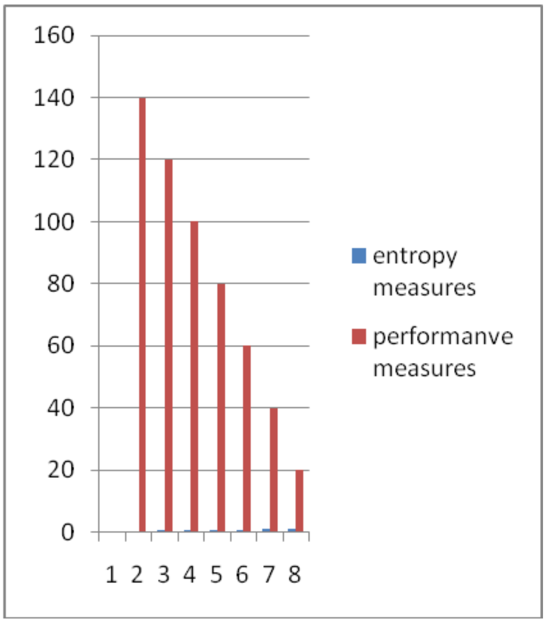


Figure 5: Network Entropy Stability Measures

REFERENCE

[AIF12] **T. E. Akhigbe-Mudu, F. T. Ibharalu, O. Folorunso** - *Analysis of Network Packet Loss Using Passive Measurement Technique Augmented with Parity Check*, Annals Computer Science Series, 10th Tome, 2nd Fasc, pages 82 – 89, 2012

[ARA12] **Alyeb Altaher, Surewaran Ramadas, and Ammar Almomani** - *Real Time Network Anomaly Detection Using Relative Entropy*, 2012.

[BCB12] **Partha Sarathi Banerjee, J. Paul Choudhury, S. R. Bhadrachaudhuri** – *A framework for Selecting the Most Reliable Path in a Computer Network Using Particle Swarm Optimization (PSO) based on Fuzzy Logic*, International Journal of Computer Applications (0975 – 8887) Volume 45, No. 8, May (2012)

[BST12] **Anup Bhange, Amber Syad, Satyendra Singh Thakur** – *DoDs Attacks Impact on Network Traffic and its Detection Approach*. International Journal of Computer Applications (0975 – 8887) Volume 40 – No. 11, page 36 – 40, February (2012)

[DRS12] **Sanjay Kumar Dubey, Ajay Rana, Arun Sharma** – *Usability Evaluation of Object Oriented Software System Using Fuzzy Logic Approach*, International Journal of Computer Applications (0975 – 8887), Volume 43, No.19, April 2012

[GB12] **Anita Garhwal, Partha Pratin Bhattacharya** – *Fuzzy Logic Based Channel Estimation and Performance Analysis of Wimax Systems*. Journal of Emerging Trends in Computing and Information Sciences. Volume 3, No. 3, page 334 – 338, March (2012)

[G+11] **Joao V. Gomes, Pedro R. M. Inacio, Manela Pereira, Mario M Freire, Paulo P. Monteiro** – *Identification of Peer – To – Peer VoIP Sessions Using Entropy and Codec Properties*. IEEE Transactions on Parallel and Distributed Systems, Volume x, No. x, (2011).

[Joh76] **Robert R. Johnson** - *Elementary Statistics*, 2nd Edition, Duxbury Press, North Scituate, Massachusetts. Pages 372 – 384, 1976.

[OCI13] **Ihekweaba Ogechi, Ihekweaba Chukwugoziem, Inyama H.C.** – *Fuzzy Modeling of a Network Denial of Service (DOS) Attack Phenomenon*, International Journal of Engineering and Technology (IJET), Vol. 5, No. 2, page 1794 – 1855, April (2013)

[RV12] **Nallagaria Ramamurthy, Dr. S. Varadarajan** – *Robust Digital Image Water Marking Scheme with Neural Network and Fuzzy Logic Approach*. International Journal of Emerging Technology and Advanced Engineering. Volume 2, Issue 9, page 555 – 562, September (2012).

[RMS08] **M.F.Rohani, M.A. Maarof, A. Selamat** – *Continuous Loss Detection Using Iterative Window Based on SoSS Model and MLS Approach*. In Proceedings of the International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, May 2008.

Tibiscus

[RTP12] **V. Ramesh, Dr. T. Stephen Thangaraj, J. V. Prasad –** *An Efficient Path Loss Prediction Mechanism in Wireless Communications Network Using Fuzzy Logic*. Internal Journal of Advanced Research in Computer Science and Software Engineering. Volume 2, Issue 1, January (2012)

[SNB04] **Shibin Song, Ng J. K.-Y., Bihai Tang –** *Some Results on the Self-Similarity Property in Communication Networks*. IEEE Transactions on Communications 52(10): 1636 – 1641, 2004.

[SSK12] **Srinivas Sethi, Anupama Sahu, Suvendu Kuma Jena –** *Efficient Load Balancing in Cloud Computing Using Fuzzy Logic*. (IOSRJEN), Volume 2, Issue 7, page 65 – 71, July (2012)

[S+11] **Hadi Shirouyehzad, Hamidreza Panjehfouladgaran, Reza Dabestani, Mostafa Badakhshian -** *Vendor Performance Measurement Using Fuzzy Logic Controller*. The Journal of Mathematics and Computer Science, Vol. 2, No. 2, page 311 – 318, (2011)

[WY08] **J.T. Wang, G. Yang –** *An Intelligent Method for Real – Time Detection of DDoS Attack Based on Fuzzy Logic*. Journal of Electronics (China), 25(4): 511 – 518, 2008.

[ZSJ10] **Zhengmin Xia, Songnian Lu, Jianhua Li –** *Enhancing DDos Attack Detection Via Intelligent Fuzzy Logic*, *Informatica* 34 (2010), pages 497 – 507.

Tibiscus