Self-similar Property for TCP Traffic under the Bottleneck Resource Restrainment

Takuo Nakashima and Akari Ono
Department of Information Science, Kyushu Tokai University
9-1-1 Toroku, Kumamoto, Japan
{taku@ktmail, 30iis111@stmail}.ktokai-u.ac.jp

Abstract

The scale-invariant burstiness or self-similarity has been found in real network. Relation between self-similarity and network and/or system parameter is mainly discussed in the context of application layer, and this self-similarity is caused by the file size of Web servers or the duration of user sessions. In addition, self-similar burstiness has been characterized by the transport layer protocol. Traffic dynamics, however, are mainly generated by physical conditions such as the resource restrainment. In this paper, we have investigated self-similar property for TCP traffic under the resource restrainment to make the network parameter vary using the network simulator. After analyzing the simulated results, following properties were extracted. Firstly, the self-similar property becomes less distinctive for the bottleneck bandwidth restrainment especially under the condition at exceeding 4 % packet loss rate. Secondly, on the saturated condition with nearly no packet loss caused by the increase of client number, the self-similar property is sustained with the Hurst parameter with the value of 0.5 for all α's. Finally, large link delay exceeding 700 msec emphasizes the self-similar property for all α's enlarging the Hurst parameter approximately at the value of 0.8.

1 Introduction

Since the seminal study of Leland, et al. [1], the scale-invariant burstiness or self-similarity has been found in real network. Currently, self-similarity of network traffic has been widely adopted in the modeling and analysis of network performance. Relation between self-similarity and network and/or system parameter is mainly discussed in the context of application layer. Crovella et al. [2] indicate that this self-similarity is mainly caused by the file size of Web servers or the duration of user sessions, and ftp traffic has the heavy-tailed property of Pareto distribution with $0.9 \leq \alpha \leq 1.1$[3].

In this paper, we focus on the self-similar property for TCP traffic under bottleneck resource restrainment. In general, the healthy Internet maintains the low error rate below one percentage, whereas some edge nodes of small provider temporally rise the error rate under the heavy load condition. These error-prone traffics are controlled by the flow control mechanism of TCP. It is the ongoing topic to investigate how this feedback mechanism effectuates the property of self-similarity. We exploit the variation of the Hurst parameter in terms of the bandwidth restrainment, the increase of the number of clients and the expansion of delay.

Park et al. [4] examined simulations with diverse conditions using network simulator, and found that the presence of self-similarity at the link and network layer depends on whether reliable and flow-controlled communication is employed at the transport layer. In the absence of reliability and flow control mechanisms, such as when a UDP-based transport protocol is used, they found that much of the self-similar burstiness of the traffic was destroyed. This result infers that the discussion of the self-similar property for TCP traffic can make the influence of the self-similarity more clarify than that of UDP traffic.

In this paper, we focus on the condition of resource restrainment for TCP traffic, and investigate the property of self-similar traffic in detail to make the network environment vary using the network simulator (ns-2) [5]. Firstly, we capture values of parameter on the condition of resource restrainment, and clarify mechanisms of TCP. Then, the discussion of Hurst parameter will be presented.

This paper is organized as follows. First, the background of self-similar property and estimation of Hurst parameter is described in Section 2 followed by the environment of network model and simulation in Section

3. With this simulation, we discuss the effect of self-similar traffic to make the network environment vary in Section 4 and Section 5. Section 6 is the summary and discussion for future work.

2 Background

We will shortly introduce the basic concepts relating to self-similarity and a more detailed treatment can be found in Park et al. [7] and Abry et al.[8].

2.1 Self-similarity and Long-Range Dependence

A process $X(t)$ is said to be self-similar, with self-similarity parameter $H > 0$, if

$$\{X(t), t \in R\} =_d \{c^H X(t/c), t \in R\}, \quad \forall c > 0, \quad (1)$$

where $=_d$ means equality for all finite dimensional distributions.

For applications, the class of self-similar processes are usually restricted to that of self-similar processes with stationary increments (or H-sssi processes). For a H-sssi process, the self-similarity parameter necessarily falls in $0 < H < 1$ and the autocovariance function $\gamma(k)$, when it exists, takes a specific, unique, and constrained form:

$$\gamma(k) = \frac{\sigma^2}{2}(|t|^{2H} - |t-s|^{2H} + |s|^{2H}). \quad (2)$$

Let $r(k) = \gamma(k)/\sigma^2$ denote the autocorrelation function. For $0 < H < 1, H \neq \frac{1}{2}$, it holds

$$r(k) \sim H(2H-1)k^{2H-2}, \quad k \to \infty. \quad (3)$$

In paticular, if $\frac{1}{2} < H < 1, r(k)$ asymptotically behaves as $ck^{-\beta}$ for $0 < \beta < 1$, where $c > 0$ is a constant, $\beta = 2 - 2H$, and we have

$$\sum_{k=-\infty}^{\infty} r(k) = \infty. \quad (4)$$

That is, the autocorrelation function, which is the essential property that causes it to be not summable, decays slowly. When $r(k)$ decays hyperbolically such that condition (4) holds, we call the corresponding stationary process $X(t)$ long-range dependence. $X(t)$ is short-range dependence if the autocorrelation function is summable.

2.2 Heavy-tailed Distribution

A random variable Z has a heavy-tailed distribution for some positive constant c, if

$$P[Z > x] \sim cx^{-\alpha}, \quad 0 < \alpha < 2, \quad (5)$$

where $a(x) \sim b(x)$ means $\lim_{x\to\infty} a(x)/b(x) = 1$, and α is called the tail index or shape parameter. Regardless of the behavior of the distribution for small values of the random variable, it is heavy-tailed if the tail of the distribution asymptotically decays under the manner of power law. If $P[Z > x]$ is heavy tailed then Z shows very high variability. Z has infinite variance for $0 < \alpha < 2$, and, if $0 < \alpha \leq 1$, Z has infinite mean. In the networking context, we will be primarily interested in the case $1 < \alpha < 2$.

The simplest heavy-tailed distribution is the Pareto distribution. The Pareto distribution is power law over its entire range; its probability density function is given by

$$p(x) = \alpha k^\alpha x^{-\alpha-1}, \quad \alpha, k > 0, \quad x \geq k, \quad (6)$$

and its cumulative distribution function is given as

$$F(x) = P[Z \leq x] = 1 - (k/x)^\alpha. \quad (7)$$

The parameter k represents the smallest possible value of the random variable, and is called the location parameter.

2.3 Estimation of Hurst Parameter

A quantitative measure of self-similarity is obtained by using the Hurst parameter H, which expresses the speed of decay of a time series' autocorrelation function. A H-sssi process with long-range dependence has an autocorrelation function of the form

$$r(k) \sim ck^{-\beta} \qquad k \to \infty, \quad (8)$$

where $c > 0$ is a constant, and $0 < \beta < 1$. The Hurst parameter is related to β via

$$H = 1 - \frac{\beta}{2}. \quad (9)$$

Heavy-tailed property causes the long-range dependence that is one of the properties of self-similarity. Park[7] presented that heavy tails lead to predictability, and in relation, they lead to long-range dependence in network traffic. Hurst parameter is related to the tail index by $H = (3-\alpha)/2$, which can be predicted by the on/off model in an idealized case corresponding to a fractal Gaussian noise process ranging over $1/2 < H < 1$ in corresponding to $1 < \alpha < 2$.

Beran[6] introduced the method, i.e. the variance-time plot, to estimate Hurst parameter. The variance-time plot is based on the slowly decaying variance of a self-similar time series, and it is expected to be scattered around a straight line with negative slope $2H-2$. This slope can be estimated based on the least squares method.

3 Network Model and Simulation

3.1 Network Model

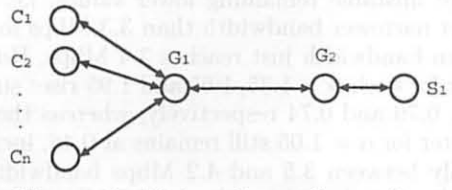

Figure 1. Network configuration.

The network is constructed by nodes and links. Each link has a buffer, bandwidth, and latency. Server node s_1 has a probability density functions $p_i(X)(i = 1, \cdots, n)$ for each TCP connection, where $X \geq 0$ is a random variable denoting file size. Client node $c_i(i = 1, \cdots, n)$ independently accesses to the server, where each request is completely determinded by the file size.

Figure 1 shows 1-server, n-client network configuration. The link between gateway G_1 and G_2 forms a bottleneck link. We will refer to the traffic from G_2 to G_1 as downstream traffic and the one from G_1 to G_2 as upstream traffic. Downstream traffic denotes multiplexed file transmission from the server.

Crovella et al. [2] indicate that this self-similarity is mainly caused by the file size of Web servers or the duration of user sessions. In addition , we measured file size of the top page on diverse Web servers and observed heavy-tailed distribution of file sizes[9]. We assume that these actual file sizes are applied to the probability density function $p_i(X)$.

3.2 Simulation Setup

We adopt the LBNL network simulator (ns) [5] to evaluate the effect of network parameters. The ns is a very popular software for simulating advanced TCP/IP algorithms and protocols. In this paper, we discuss the relation between self-similar property and the variation of restricted bandwidth.

We conducted two different types of experiments; a bottleneck bandwidth restrainment and a bottleneck delay restrainment, discussing in later chapters. The topology of bandwidth restrainment consists of 1-sever 32-client nodes and 15 msec latency presenting domestic links with variation of bandwidth. The topology of delay restrainment consists of 1-server 128-client nodes and 10 Mbps bandwidth with variation of delay. Non-bottleneck links are all set at 10 Mbps bandwidth and 15 msec latency. The maximum segment size is fixed at 1 kbytes. We measure the downstream traffic from G_2 to G_1.

4 Effects of Bottleneck Restrainment

4.1 Effects of Bandwidth Restrainment

We examine bottleneck link restraining bandwidth between G_1 and G_2. Self-similar traffic is generated by the file size on the server with Pareto distribution, i.e. heavy-tailed distribution. As the shape of Pareto distribution is mainly determined by the shape parameter(α), we conduct four different values, i.e. $\alpha = 1.05, 1.35, 1.65$, and 1.95. In general, if α is close to 2, the process sequence has the property of short-range dependence and operates similar to the process with exponential distribution. On the other hand, if α is close to 1, the process sequence has the property of long-range dependence. We set the packet size to 1 kbyte, data traffic rate to 200 kbps. Each TCP agent connected to respective client TCPsink agent is set to Reno and the window size of TCP is set to 100.

4.1.1 Throughput Variation

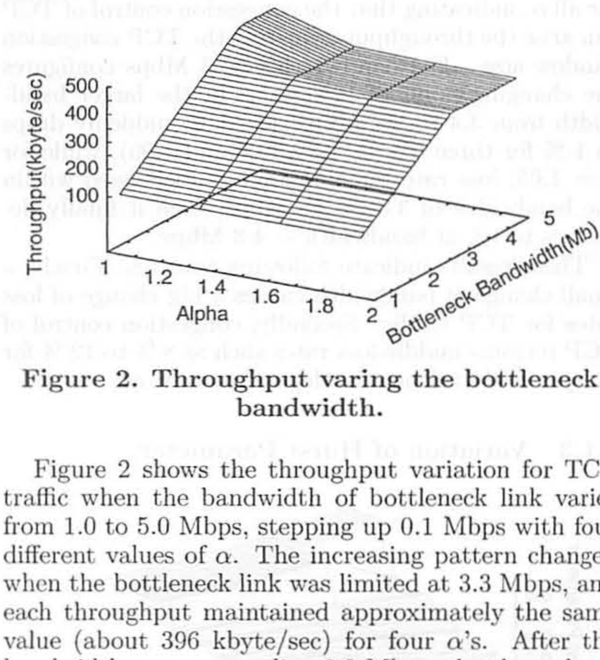

Figure 2. Throughput varing the bottleneck bandwidth.

Figure 2 shows the throughput variation for TCP traffic when the bandwidth of bottleneck link varies from 1.0 to 5.0 Mbps, stepping up 0.1 Mbps with four different values of α. The increasing pattern changed when the bottleneck link was limited at 3.3 Mbps, and each throughput maintained approximately the same value (about 396 kbyte/sec) for four α's. After the bandwidth grows exceeding 3.3 Mbps, the throughput curve at $\alpha = 1.05$ increases with two slopes. Firstly, the throughput gradually increases until bandwidth = 4.3 Mbps, then is maintained the same value. The throughput for smaller α is slightly larger than that for larger α. After releasing the restrainment of bottleneck, smaller α contributes to raise the throughput performance, indicating that the property of self-similarity enlarges the throughput.

4.1.2 Loss Rate Variation

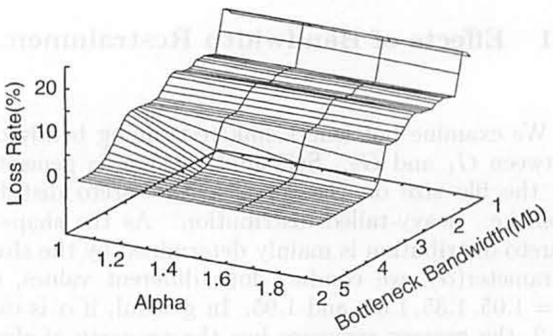

Figure 3. Loss rate varing the bottleneck bandwidth.

Figure 3 shows the loss rate variation of bottleneck restrainment for TCP traffic. To show the descending pattern of loss rate without overlapped lines, the value of order of bandwidth axis is changed to the descending order. At small bandwidth from 1.0 to 1.3 Mbps, limited bandwidth generates traffic congestion, leading to large loss rate. Under the condition of smaller bandwidth, that is under 3.2 Mbps, loss rate roughly remains from 9 to 12 %, and maintains the similar value for all α, indicating that the congestion control of TCP can arise the throughput adjusting the TCP congestion window size. The bandwidth = 3.3 Mbps configures the changing point of loss rate. In the larger bandwidth from 3.4 to 3.7 Mbps, loss late suddenly drops to 1 % for three α's (= 1.35, 1.65 and 1.95), while for $\alpha = 1.05$, loss rate remains almost unchanged within the bandwidth of 3.3 to 3.8 Mbps, then it finally decreases to 1% at bandwidth = 4.3 Mbps.

These results indicate following features. Firstly, a small change of bandwidth causes a big change of loss rates for TCP traffic. Secondly, congestion control of TCP remains middle loss rates such as 8 % to 12 % for every α and vanishes the dependency for α.

4.1.3 Variation of Hurst Parameter

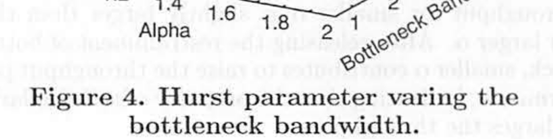

Figure 4. Hurst parameter varing the bottleneck bandwidth.

Figure 4 illustrates fluctuations of the Hurst parameter to make bottleneck bandwidth vary. These Hurst parameters are estimated by the slope of the variance-time plot for the throughput based on the least squares method. In the restricted bandwidth, the Hurst parameters are unstable remaining lower values, i.e. under 0.6, over narrower bandwidth than 3.3 Mbps for every α. When bandwidth just reaches 3.4 Mbps, Hurst parameter for each $\alpha = 1.35, 1.65$ and 1.95 rises suddenly to 0.65, 0.79 and 0.74 respectively, whereas the Hurst parameter for $\alpha = 1.05$ still remains at 0.46, increasing gradually between 3.5 and 4.2 Mbps bandwidth. Finally, when bandwidth reaches at 4.2 Mbps, the Hurst parameter for each $\alpha = 1.05, 1.35, 1.65$ and 1.95 stays in 0.91, 0.75, 0.66 and 0.63 respectively, moving to the stable values for each α.

As the results, we can extract the following. Hurst parameters for TCP traffic suddenly change from unstable state to stable state for larger α's(=1.35, 1.65 and 1.95), synchronizing the changing point of loss rate from over 8 % to under 4 %. For the small α=1.05, Hurst parameter converges to stable state when loss rate decreases under 4 %. These results mean that the property of self-similarity for TCP traffic appears under the loss rate = 4 %.

Park et al. [4] conducted the simulation using ns-2, and compared the estimated values of Hurst parameter for limited bandwidth (1.5 Mbps) and non-limited bandwidth (155 Mbps). They concluded that bandwidth limitation led to no significant variations for self-similarity. Park et al. [4] also described that limited bandwidth resulted in 4 % packet drop rate for the most bursty case (α=1.05). Compared to our results, the condition of the limited bandwidth (1.5 Mbps) has already converged to the stable state, and the results could not lead to the difference.

In TCP, 4 % packet loss rate is the changing point from unstable and stable state for the estimated Hurst parameter. After reaching the stable state, a Hurst parameter is maintained in a linear manner for each α. It can be also said that bottleneck bandwidth restrainment vanishes the property of self-similarity.

4.2 Effects of the number of Client

In the previous subsection, we concluded heavy loss rate over 4 % led to unstable Hurst parameters and vanished the self-similar property. In this and next subsections, we will burden the bottleneck link with the number of client and delay without occurring the packet loss.

4.2.1 Throughput Variation

Firstly, we increase the number of clients. To remove the influence of the packet loss, the bandwidth of bottleneck link will be increased to 10 Mbps and the delay will be set to 15msec.

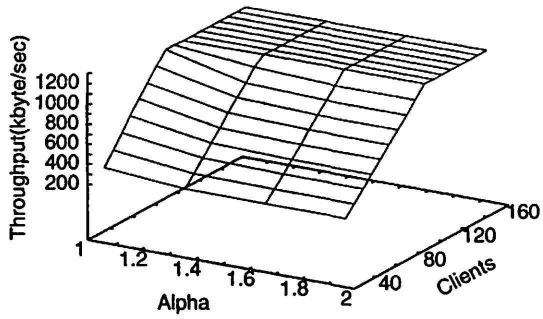

Figure 5. Throughput varying the number of client.

Figure 5 illustrates the throughput variation when the number of client varies from 24 to 150 with 8 step. The pattern, forming smaller α that generates larger throughput, was kept under about 88 clients. If the number of clients increases over 96, then the throughput maintains the same value for each α. The larger number of client causes the congestion control of TCP remaining the best throughput on this condition, while packet loss hardly appeared in all cases. This results means that the large number of clients vanish the characteristic of throughput pattern for different α's.

4.2.2 Variation of Hurst Parameter

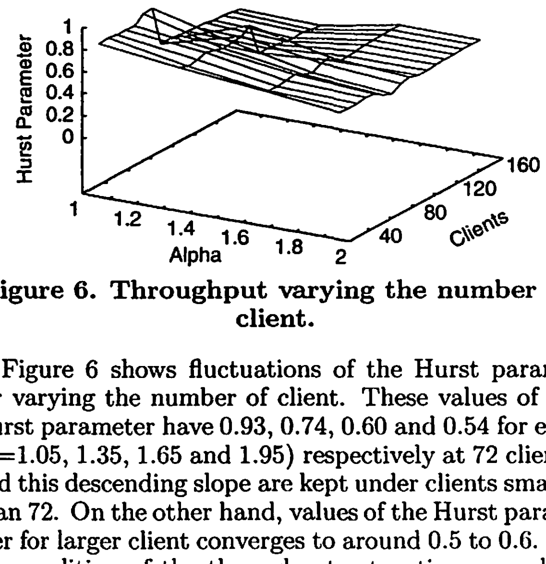

Figure 6. Throughput varying the number of client.

Figure 6 shows fluctuations of the Hurst parameter varying the number of client. These values of the Hurst parameter have 0.93, 0.74, 0.60 and 0.54 for each α(=1.05, 1.35, 1.65 and 1.95) respectively at 72 clients, and this descending slope are kept under clients smaller than 72. On the other hand, values of the Hurst parameter for larger client converges to around 0.5 to 0.6. On the condition of the throughput saturation caused by the increase of client number, the self-similar property will become less distinctive like the bottleneck bandwidth restrainment.

4.3 Effects of Delay Restrainment

In previous subsection, the traffic of 72 clients configured the border value whether the self-similar property is maintained or not. In this subsection, we will fix the number of client to 64 which is close to the boundary value of 72, and make the bottleneck delay variable from 100 msec to 2000 msec. As all simulated data under 100 msec delay is likely to have the similar features in pre-experiments, we have conducted the delay is to exceeded 100 msec.

4.3.1 Throughput Variation

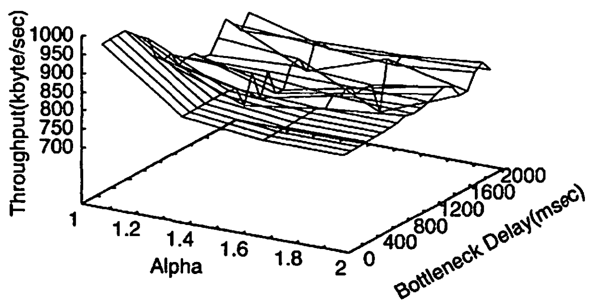

Figure 7. Throughput varying the bottleneck delay.

Figure 7 shows the throughput variation for TCP traffic when the delay of bottleneck link varies from 100 to 2000 msec stepping up 100 msec with four different values of α. The declining line of the Hurst parameter, indicating the throughput of smaller α's hold larger values of the Hurst parameter, maintained approximately between 100 msec and 2000 msec. Among the last part of increasing link delay up to 2000 msec, the throughput values fluctuated irregularly. The result indicates that the flow control of TCP could not converge the throughput to the smooth variation over larger link delay exceeding 700 msec.

4.3.2 Variation of Hurst Parameter

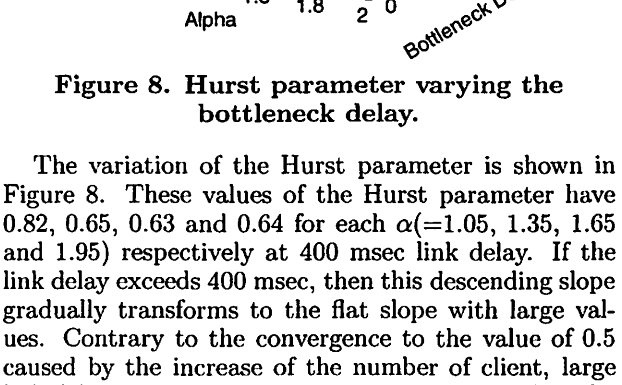

Figure 8. Hurst parameter varying the bottleneck delay.

The variation of the Hurst parameter is shown in Figure 8. These values of the Hurst parameter have 0.82, 0.65, 0.63 and 0.64 for each α(=1.05, 1.35, 1.65 and 1.95) respectively at 400 msec link delay. If the link delay exceeds 400 msec, then this descending slope gradually transforms to the flat slope with large values. Contrary to the convergence to the value of 0.5 caused by the increase of the number of client, large link delay causes the Hurst parameter for all α's to be approximately at the value of 0.8. This result indicates that the large link delay enlarges the property of self-similarity. On the other hand, increasing the number of clients suppresses the self-similar property especially for small α's(=1.05 and 1.35). In addition, if the flow control mechanism of TCP does not operate properly, then the self-similar property is enlarged.

5 Conclusion

In this paper, we have investigated the property of self-similar traffic for TCP traffic under the resource restrainment to make the network parameter vary using the network simulator. After analyzing the simulated results, following properties were extracted. Firstly, the self-similar property becomes less distinctive for the bottleneck bandwidth restrainment especially under the condition at exceeding 4 % packet loss rate. Secondly, on the saturated condition with nearly no packet loss caused by the increase of client number, the self-similar property is sustained with the Hurst parameter with the value of 0.5 for all α's. Finally, large link delay exceeding 700 msec emphasizes the self-similar property for all α's enlarging the Hurst parameter approximately at the value of 0.8.

These simulations were examined by the variation of bottleneck bandwidth and delay. To extract network parameters effecting the property of self-similarity, other simulations restricting network resources are required. In the next step, we will conduct the simulation to make network parameters vary such as queue length. In the future, the real network experiment is to be conducted followed by discussions of the precise network simulator to find the new metrics for the self-similarity.

References

[1] Leland, W., Taqqu, M., Willinger, W. and Wilson, D.: On the Self-Similar Nature of Ethernet Traffic (Extended Version), *IEEE/ACM Trans. on Networking*, Vol.2, No.1, pp.1-15, February (1994).

[2] Crovella, M and Bestavros, A: Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes, *IEEE/ACM Trans. on Networking*, Vol.5, No.6, pp.835-845 (1997).

[3] Paxson, V. and Floyd, S.: Wide-Area traffic: the failure of Poisson modeling, *IEEE/ACM Trans. on Networking*, Vol.3, No.3, pp.226-244 (1995).

[4] Park, K. ,Kim, G. and Crovella, M. E.: The Protocol Stack and Its Modulating Effect on Self-similar Traffic, *Self-Similar Network Traffic and Performance Evaluation*, K.Park and W.Willinger,Eds.,Wiley-Interscience, New York, pp.349-366, (2000).

[5] UCB/LBNL/VINT groups. UCB/LBNL/VINT Network Simulator, http://www.isi.edu/nsnam/ns/, May (2001).

[6] Beran, J.: *Statistics for Long-Memory Processes.*, Monographs on Statistics and Applied Probability. Chapman and Hall, New York, (1994).

[7] Park, K. ,Willinger, W.: Self-similar Network Traffic: An Overview *Self-Similar Network Traffic and Performance Evaluation*, K.Park and W.Willinger,Eds.,Wiley-Interscience, New York, pp.1-38, (2000).

[8] Abry, P., Baraniuk, R., Flandrin, P., Riedi, R., and Veitch, D.: The Multiscale Nature of Network Traffic: Discovery, Analysis, and Modelling, *IEEE Signal Processing Magazine*, pp.28-46, May, (2002).

[9] Nakashima, T.,Tsuichihara, M.: Stability in Web Server Performance with Heavy-Tailed Distribution, *Proc. of 9th International Conference of Knowledge-Based Intelligent Information and Engineering Systerms(KES 2005)*, Vol. 1, pp.575-581, (2005).