High-Speed Calculation Method of the Hurst Parameter Based on Real Traffic

Tatsuya Hagiwara, †Hiroki Doi, Hideki Tode, and, ††Hiromasa Ikeda

Graduate School of Engineering, Osaka University
2–1, Yamadaoka, Suita-shi
Osaka 565-0871, Japan
E-mail: hagiwara@comm.eng.osaka-u.ac.jp
E-mail: tode@ise.eng.osaka-u.ac.jp

†Central Research Institute of Electric Power Industry
2-11-1, Iwado-kita, Komae-shi
Tokyo, 201-8511, Japan
E-mail:doi@criepi.denken.or.jp

††Tokyo University of Information Sciences
1200-2, Taniatari-chou, Chiba-shi
Chiba, 265-8501, Japan
E-mail:ikeda@rsch.tuis.ac.jp

Abstract

Recent studies on traffic measurement analysis in the various networks have shown that packet traffic exhibits Long Range Dependent properties called Self-Similarity. Some papers reported that Self-Similarity degrades the network performance, such as buffer overflow. Thus, we need new network control considering Self-Similar properties. Network control considering the Self-Similarity requires high speed calculation method of the Hurst Parameter. However, such method has not been proposed yet. In this paper, we propose high-speed calculation method of the Hurst Parameter based on the Variance-Time Plot method, and show its performance. Furthermore, we try to show effectiveness of the network control with Self-Similarity.

1 Introduction

Recently, as internet users increase, the network congestion happens frequently. As a way of moderating the network congestion, it is thought to replace existing networks with mass capacity networks. However, it is not realistic, thus, effective use of the limited network resource is needed and the behavior of network traffic is paid attention to for the basic network control technology. Up to now, as for various network designs, it has been assumed that network traffic randomly arrives. Recent studies on packet network measurement analysis in various networks[1][2] [3], however, have shown that packet traffic exhibits Long Range Dependent (LRD) properties[4][5][6] which means large variance and Self-Similarity[7]. Therefore, the traffic in the current multimedia networks such as the internet, has a tendency like that a traffic follows not the random arrival but continuous arrival with a certain bias. Accordingly, the arrival of traffic will concentrate in a short time and the buffer will overflow more frequently. As a result, the network utilization decreases remarkably due to retransmitting the overflowed packets. Like this, Self-Similar traffic degrades the network performance controlled by the conventional systems [8]. On the contrary, we will be able to use the network resource more effectively, if networks are controlled considering Self-Similar properties. However, such network control has hardly been proposed yet, and only a few proposed network control [9][10][11] requires that the Hurst Parameter which is typical measure representing the Self-Similarity be constant. Actually, the Hurst Parameter(H) changes frequently in real traffic. That is, it is difficult to apply these methods to the equipment in the real traffic environment(i.e., LAN,MAN,WAN) directly. In addition, because it takes huge time to calculate H, it is also difficult to apply a current calculation method to an actual network control. That is, it is necessary to calculate the Hurst Parameter in real time for various network controls considering Self-Similarity in the real traffic environment. In this paper, we propose On-Time VTP(Variance Time Plot) method based on the VTP [12] which is most general Hurst Parameter calculation method, and show its performance. Furthermore, we try to apply this method to the simple network control, in order to show effectiveness of the network control with Self-Similarity. The rest of the paper is organized as follows. In section 2, we briefly introduce the definition of Self-Similarity and the calculation method of the Hurst Parameter. In section 3, we estimate the computational cost of the conventional calculation method of the Hurst Parameter. In section 4, we propose the High Speed Calculation Method of the Hurst Parameter(On-Time VTP). In section 5, we evaluate the On-Time VTP based on real traffic, and apply to the simple network control with Self-Similarity. Finally, in section 6 we conclude this paper, and discuss future work on the topic of the network control considering Self-Similarity.

2 Definition of Self-Similarity

In this paper, the Self-Similarity is focused on as for variance trend of traffic. We take variance time plot method to evaluate the Self-Similarity of traffic and its Hurst Parameter.

Let $X = (X_t : t = 0, 1, 2, \ldots)$ be a covariance-stationary stochastic process with mean $\mu = E(X_t)$, variance $\sigma^2 = Var(X_t)$ and autocorrelation function $\gamma(k)$. $\gamma(k)$ is given by

$$\gamma(k) = Cov(X_t, X_{(t+n)})/\sigma^2, \quad k = 0, 1, 2, \ldots$$

In this paper, this process means packets arrival process. X_t is the volume(*bytes*) of arrival packets at the t-th time slot (time slot = $10ms$. in this paper). For each $m = 1, 2, 3, \ldots$, let $X^{(m)} = (X_k^{(m)} : k = 1, 2, 3, \ldots)$ denote the new covariance stationary time series obtained by averaging the original series X non-overlapping blocks of size m. That is, for each $m = 1, 2, 3 \ldots$, $X_k^{(m)}$ is given by

$$X_k^{(m)} = \frac{1}{m}(X_{km-m+1} + \cdots + X_{km}), \quad k \geq 1 \tag{1}$$

In this case, for all $m = 1, 2, \ldots$, the time series $X^{(m)}$ is the same covariance stationary stochastic process as the time series X, and mean, variance and autocorrelation function are respectively given by

$$\begin{aligned} E(X^{(m)}) &= \mu, \\ Var(X^{(m)}) &= \frac{\sigma^2}{m} + \frac{2\sigma^2}{m^2}(m-j)\gamma(j), \\ \gamma^{(m)}(k) &= \frac{\sigma^2}{m^2 Var(X^{(m)})}\{m\gamma(m \cdot k) \\ &+ 2\sum_{j=1}^{m-1}(m-j)\gamma(m \cdot k + j)\}. \end{aligned}$$

If $X^{(m)}$ has the same Self-Similar structure as X, the following expressions hold,

$$\begin{aligned} \gamma^{(m)}(k) &= \gamma(k) \\ Var(X^{(m)}) &= \sigma^2 m^{-\beta} \end{aligned}$$

In this case, the process X is called exactly second-order Self-Similar with the Hurst Parameter $H = 1 - \beta/2$. In this paper, we adopt the Hurst Parameter as an evaluation index of the Self-Similarity. If m is large enough, we can define asymptotical Self-Similarity. The asymptotical Self-Similarity is often used in the field of network traffic theory, because it is more practical. In this case, Self-Similarity is defined as follows.

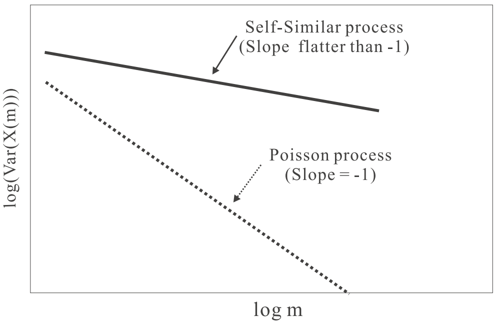

Figure 1: Self-Similar process and Poisson process

$$Var(X^{(m)}) \sim c\,m^{-\beta}, \; as\; m \to \infty, \quad c : const$$

That is, the variance of the sample mean decreases more slowly than the reciprocal of the sample size m. Fig.1 shows this process as m is increasing.

There are several kinds of calculating method of the Hurst Parameter. The typical calculating methods are as follows.

- Variance Time Plot(VTP)
- R/S Analysis
- Periodgram
- Whittle's Estimator

In this paper, especially, we focus on the Variance Time Plot and try to improve it, because VTP obtains the Hurst Parameter faster and more easily than the other methods and we can directly see Self-Similar phenomenon from result. However, even this method requires about an hour to calculate Hurst Parameter from the data set(3200sec length, described in Table 1) by the *PC/AT Compatible Computer* (Pentium 233MHz, 64MB RAM) for example. Thus for, this was not severe problem because main objections to calculate Hurst Parameter was to just analyze the long time trend of the target traffic. Therefore, it is difficult to apply this method to actual network control directly.

First, we describe outline of the VTP. Let $X = (X_t : t = 0, 1, 2 \ldots)$ be time series of a volume(*bytes*) of arrival packets. We calculate $X^{(m)}$ according to eq.(1), plot

$$f(m) = (log(m), log((Var(X^{(m)})))),$$

and obtain the following value(see Fig.2),

$$f'(target) = -\beta_{target}.$$

Finally, we can get the Hurst Parameter from following equation

$$H_{target} = 1 - \beta_{target}/2.$$

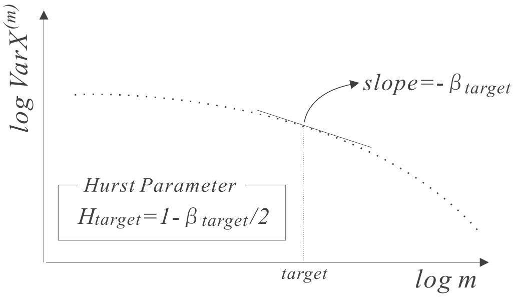

Figure 2: Variance Time Plot

Next, we will explain the procedure of calculating the Hurst Parameter by the VTP due to simple example.

It is assumed that $X^{(1)}$ is the collecting network packets with 10ms time slot, and shown as follows,

$$X^{(1)} = (2, 7, 4, 12, 5, 0, 8, 2, 8, 4, 6, 9, 1 \cdots).$$

When $X^{(1)}$ is as prepared, we calculate $X^{(2)}$ according to eq.(1). $X^{(2)}$ is given by

$$X^{(2)} = (4.5, 8, 2.5, 5, 6 \cdots).$$

We call this phase **Phase of making series (P_{ser})**. When derivation of $X^{(k)}$ $(1 \leq k \leq m)$ finished, we calculate the following variance,

$$Var(X^{(1)}), Var(X^{(2)}), \cdots, Var(X^{(m)}).$$

We call this phase **Phase of calculating variance (P_{var})**. By this two phases

$$(log(m), log(Var(X^{(m)})))$$

is calculated, and we can get the Hurst Parameter as shown before.

3 Computational Cost of Estimation of the VTP

To speed up the VTP, we consider the computational cost of each phase of the VTP. We assumed the following conditions.

- The number of events processed by the VTP is N.
- One arithmetic operation requires unit computational cost.
- $2 \leq k \leq N/2$.

The computational cost of P_{ser} is defined as C_{ser}. P_{ser} is composed of addition(+) and division(÷) operations. If

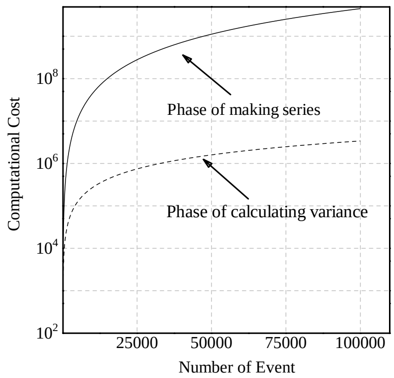

Figure 3: Computational Cost

$k = 2$, computational costs of additions and divisions are respectively given by

$$\begin{aligned} Computational\ Cost\ of \text{ “ + ” } &= \left\lceil \frac{N}{2} \right\rceil, \\ Computational\ Cost\ of \text{ “ ÷ ” } &= \left\lceil \frac{N}{2} \right\rceil. \end{aligned}$$

Therefore, $C_{ser:k=2}$ is given by

$$C_{ser:k=2} = 2 \left\lceil \frac{N}{2} \right\rceil.$$

For each $k = 2, 3, \ldots, N/2$, we can calculate C_{ser}. C_{ser} is given by

$$\begin{aligned} C_{ser} &= 2 \left\lceil \frac{N}{2} \right\rceil + 3 \left\lceil \frac{N}{3} \right\rceil + \cdots \\ &+ \frac{N-1}{2} \left\lceil \frac{N}{(N-1)/2} \right\rceil + \frac{N}{2} \left\lceil \frac{N}{N/2} \right\rceil \\ &= \sum_{k=2}^{N/2} k \left\lceil \frac{N}{k} \right\rceil. \end{aligned} \quad (2)$$

The computational cost of P_{var} is defined as C_{var}. P_{var} is composed of

- Process 1 – squaring all elements and averaging.
- Process 2 – calculating the sum of all elements and averaging.
- Process 3 – calculating difference between Process 1 and Process 2.

Let us start with considering Process 1's computational cost (C_{sq}). If $k = 2$, computational costs of multiplications, additions and divisions are given by

$$\begin{aligned} Computational\ Cost\ of \text{ “ × ” } &= N, \\ Computational\ Cost\ of \text{ “ + ” } &= N-1, \\ Computational\ Cost\ of \text{ “ ÷ ” } &= 1. \end{aligned}$$

Thus, $C_{sq:k=2}$ is given by

$$C_{sq:k=2} = N + N - 1 + 1 = 2N$$

Next, we consider Process 2's computational cost (C_{av}). If $k = 2$, computational costs of multiplications and divisions are given by

$$\begin{aligned} Computational\ Cost\ of\ \text{“}\times\text{”} &= N - 1, \\ Computational\ Cost\ of\ \text{“}\div\text{”} &= 1. \end{aligned}$$

Thus, $C_{av:k=2}$ is given by

$$C_{av:k=2} = N - 1 + 1 = N.$$

Therefore $C_{var:k=2}$ is given by

$$C_{var:k=2} = C_{sq:k=2} + C_{av:k=2} + 1 = 3N + 1.$$

From the above discussion, for each $k = 2, 3, \ldots, N/2$, C_{var} is expressed as follows,

$$\begin{aligned} C_{var} &= C_{sq} + C_{av} + 1 \\ &= (3N+1) + \left\{3\left\lceil\frac{N}{2}\right\rceil + 1\right\} \\ &+ \left\{3\left\lceil\frac{N}{3}\right\rceil + 1\right\} + \cdots + \left\{3\left\lceil\frac{N}{N/2}\right\rceil + 1\right\} \\ &= 3\sum_{k=1}^{N/2}\left\lceil\frac{N}{k}\right\rceil + \frac{N}{2}. \end{aligned} \quad (3)$$

The C_{ser} and C_{var} are presented in Fig.3. We can see C_{ser} is much more than C_{var}. That is, P_{ser} is bottleneck of the VTP. In this paper, we focus on this P_{ser} and improve calculating speed of the VTP.

4 High-speed calculation method of the Hurst Parameter

4.1 Basic concept

As we have described before, it is necessary to know Hurst Parameter in real time to control networks more efficiently by considering Self-Similarity. Thus, in this paper, we propose On-Time VTP[14] which can calculate the Hurst Parameter in real time and continuously while the conventional VTP calculates the Hurst Parameter only in non-real time and collectively. In this paper, we improve the following three points to the conventional VTP.

- Periodical output of the Hurst Parameter per short cycle.
- Concurrent processing (to achieve real-time processing).
- Restriction of range of calculation (to reduce computational cost).

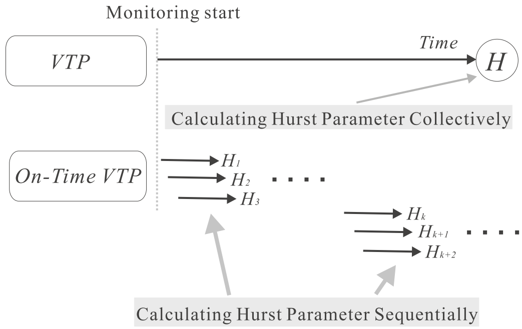

Figure 4: Output timing of the On-Time VTP

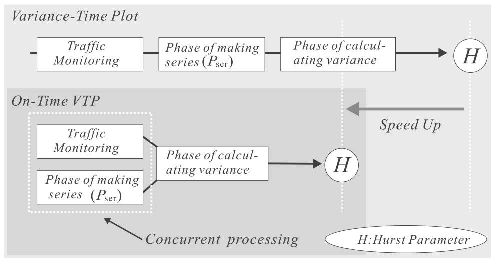

Figure 5: VTP and On-Time VTP

4.2 Periodical output of the Hurst Parameter

To control networks considering Self-Similarity, it is necessary to calculate the Hurst Parameter at a short cycle and notify the value to the control systems. However, it is difficult to apply the VTP to network control directly, because the conventional VTP can calculate the Hurst Parameter only at a long cycle. Then, to calculate the Hurst Parameter continuously at a shot cycle, we have adopted concurrent processing like Fig.4 This periodical output of the Hurst Parameter is described in detail in the following subsection.

4.3 Concurrent processing

Figure 5 shows block chart of the VTP and On-Time VTP. The VTP processes all procedures one after another. On the other hands, the On-Time VTP processes P_{ser} which requires a lot of computational cost and *Phase of Traffic Monitoring* concurrently. Thus, the calculation of Hurst Parameter becomes much faster. In this paper, *Traffic Monitoring* is that the probe (e.g. $tcpdump$) monitors the packets and records its *time stamp* and *byte*.

Here, let us think about the case where the process of calculating the Hurst Parameter and the other processes share a CPU. If the On-Time VTP process P_{ser} and *Traffic Monitoring* concurrently, the computational cost for each unit time of this part will increase. In this case, if the load of the process of calculating the Hurst Parameter is

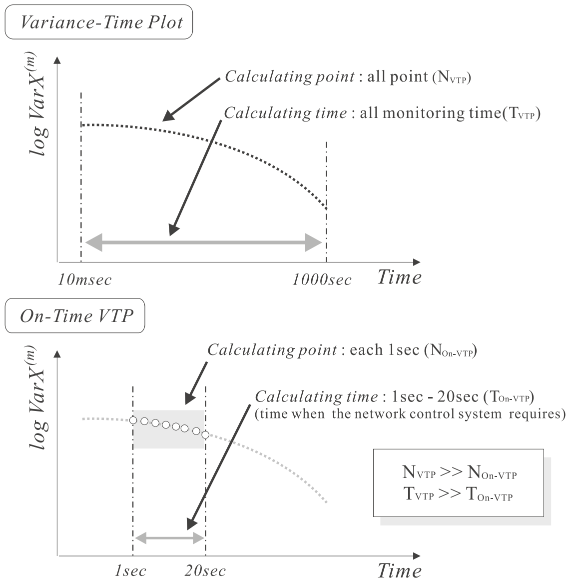

Figure 6: Restriction of range of calculation

high, performance of the other process may be degraded. Thus, we have reduced the computational cost by using the method described in section 4.4. As a result, the influence given to the CPU decreases.

4.4 Restriction of range of calculation

Actually, to control networks more efficiently considering Self-Similarity, it is necessary to know the behavior of traffic of the time scale corresponding to the target network control method. It is reported that the behavior of traffic at the time scale of several seconds influences the network performance. In this paper, therefore, we focus on the traffic behavior at several seconds. By disregarding the time scale excluding this, we have reduced the computational cost of calculating the Hurst Parameter. The VTP always calculates the Hurst Parameter every minimum time slot(in this paper $10msec$). Then, in this proposal method, we achieve the speed-up of calculating the Hurst Parameter by changing the observing time slot from $10ms$ to $1sec$. Figure 6 shows this method in detail.

Here, the computational costs of the VTP and On-Time VTP are defined as C_{VTP} and C_{On-VTP} respectively.

If the *Traffic Monitoring* time is $1000sec$, the effect of restriction of the computational cost becomes

$$\frac{C_{On-VTP}}{C_{VTP}} \sim \frac{1}{7.0 \times 10^2}.$$

From this equation, it is clear that the computational cost of the On-Time VTP is greatly reduced than that of VTP.

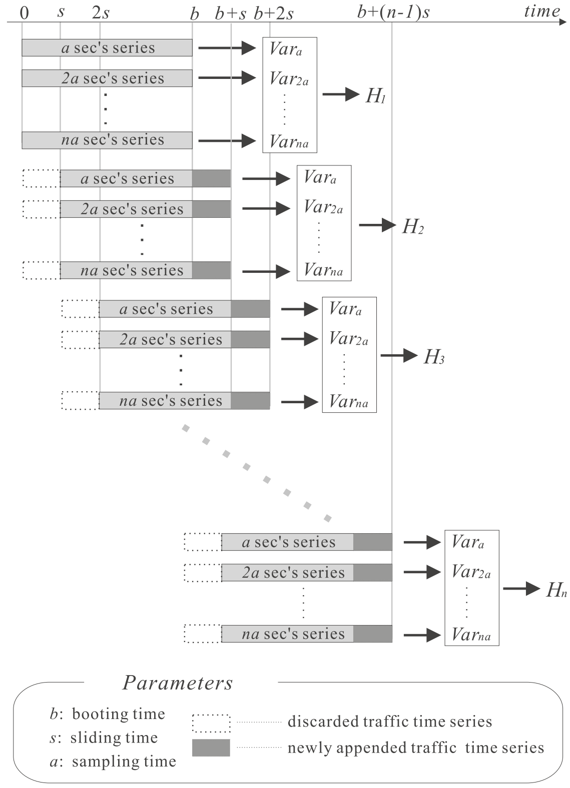

Figure 7: On-Time VTP

4.5 On-Time VTP

Up to now, we have described three speed-up techniques. Here, we propose the high-speed calculation method of Hurst Parameter (On-Time VTP) by using these techniques. First, we show the operation sequence of On-Time VTP on Fig.7. The On-Time VTP uses three parameters. The role of parameters is as follows. In the Fig.7, the parameter b($booting\ time$) means the boundary time like the following,

$[0 \leq t \leq b]$	the time from start of On-Time VTP to the first calculation of Hurst Parameter,
$[b \leq t]$	period of traffic monitoring periodically output the Hurst Parameter.

Moreover, by changing the size of b, the On-Time VTP can adjust the correlation with the past. The parameter *s(sliding time)* means unit time sliding period. Every sliding time s , the On-Time VTP outputs the Hurst Parameter and discard old block of stored traffic data. By changing size of s, the On-Time VTP can adjust the period of calculation of the Hurst Parameter. The parameter *a (sampling time)* means minimum observation time unit in order to calculate the Hurst Parameter. By changing size of a, the

On-Time VTP know behavior of the traffic at the desired time scale(*parameter a*).

Next, let's show an actual operation in detail. In $0 \leq t \leq b$, On-Time VTP makes time series

$$each\ a\ sec, \cdots, each\ na\ sec$$

monitoring traffic. This series are the same as the series which the VTP obtained in the phase of making series. However as for the On-Time VTP, the number of series obtained is much fewer than that of the VTP. Moreover, the On-Time VTP processes P_{ser} and *Traffic Monitoring* concurrently. Thus, the On-Time VTP can make n traffic series corresponding to interval b. Using this series, the series of variance

$$Var_a, Var_{2a}, \cdots, Var_{(n-1)a}, Var_{na} \tag{4}$$

is given. When time is $t = b$, the On-Time VTP calculate H_1 which is the Hurst Parameter from 0 to b. In this paper, we assume $n = 20$ and $a = 1$. When time is $b < t \leq b + s$, the On-Time VTP discards the part of data series(white part of Fig.7). Then, the On-Time VTP collects new arrival traffic during $b < t \leq b + s$ and join this series(length = s; dark gray part of Fig.7) to the storing series(length=$b - s$; bright gray part of Fig.7). Thus, the On-Time VTP can make n new time series corresponding to interval b. When time becomes $t = b+s$, as described before, the On-Time VTP makes series of variance (such as eq.(4)) and calculate H_2 which is the Hurst Parameter from s to $b+s$. Similarly, the On-Time VTP calculate Hurst Parameter series H_k ($k = 1, 2, \ldots$). Due to this concurrency, the On-Time VTP can calculate the Hurst Parameter at real time and high speed.

5 Performance evaluation

We evaluate the performance of On-Time VTP, to verify the effectiveness in the actual traffic environment. In this evaluation, we use two empirical traffic data set which contain $time$ and $byte$. These traffic data set had been collected at Bellcore in 1989[1]. We show the feature of these traffic data set on table 1.

First, we evaluate *Processing time* and *Required memory* of the On-Time VTP as basic performance, using traffic data set BC-pAug89.TL. We measure the elapsed time

Table 1: Using Traffic Data Set

	average traffic rate	length
BCpAug89.TL	1382kbps	3100sec
BC-pOct89.TL	3627kbps	1760sec

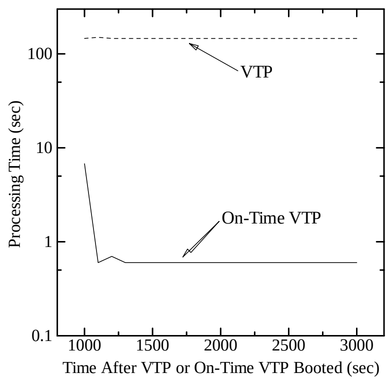

Figure 8: Processing time Performance

taken to execute the On-Time VTP and VTP, respectively, and the memory volume required for both methods. We set parameters of On-Time VTP as follows,

$$b = 1000,\ s = 100,\ a = 1.$$

The conventional VTP can't use those three parameters. Then, we cut traffic data set like Fig.7 for the conventional VTP to suitable the same condition as On-Time VTP.

Figure.8 shows the result of *Processing time*. From this result, we can see that the On-Time VTP is about 200 times faster than the VTP. In Fig.8, the initial part of the processing time of On-Time VTP is more than the other region of the processing time of it. This is due to reading traffic data from the disk. Once the On-Time VTP read the traffic data from the disk, the On-Time VTP can process very quickly. While, we find that the On-Time VTP requires less than 1MB memory, but the VTP requires more than 2MB memory. Then it is recognized that the memory required for the On-Time VTP is twice fewer than that of the VTP. Thus, if we design the network control system using the On-Time VTP, the On-Time VTP influence the network performance very slightly.

Next, we evaluate the performance of the On-Time VTP based on real traffic, using the data set shown in table 1. The evaluation index is Error rate and Sensitivity ratio. Here, the series of Hurst Parameter from the VTP is given by

$$H_{VTP}(1), \cdots, H_{VTP}(n), \cdots, H_{VTP}(N)$$

and, the series of Hurst Parameter from the On-Time VTP is given by

$$H_{On-TimeVTP}(1), \cdots, H_{On-TimeVTP}(N)$$

The *Error Rate* in the n-th Hurst Parameter is defined

$$Error\ Rate(n) = \frac{H_{VTP}(n) - H_{On-TimeVTP}(n)}{H_{VTP}(n)}.$$

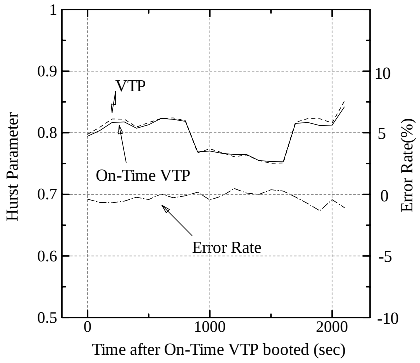

Figure 9: Error rate : BC-pAug89.TL

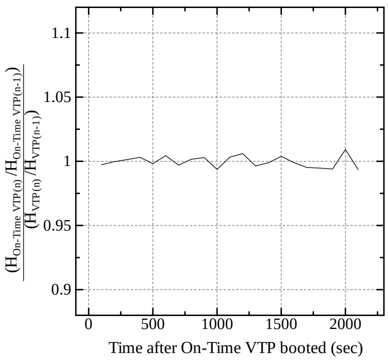

Figure 10: Sensitivity ratio : BC-pAug89.TL

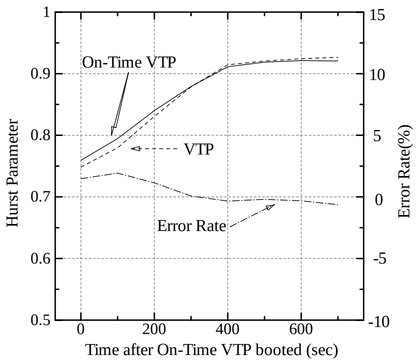

Figure 11: Error rate : BC-pOct89.TL

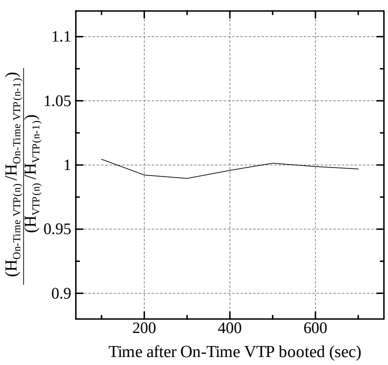

Figure 12: Sensitivity ratio : BC-pOct89.TL

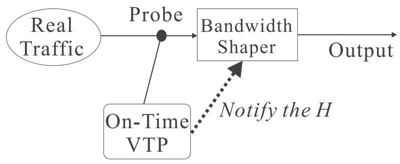

Figure 13: Simulation Environment

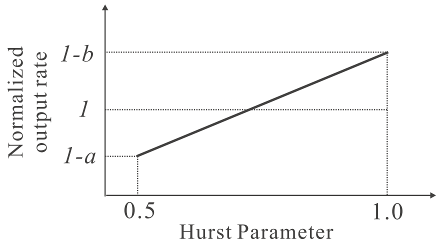

Figure 14: Bandwidth Allocation Method

and, the *Sensitivity Ratio* is defined

$$Sensitivity\ Ratio = \frac{H_{On-TimeVTP}(n)/H_{On-TimeVTP}(n-1)}{H_{VTP}(n)/H_{VTP}(n-1)}.$$

If the error rate is low and sensitivity ratio is also near 1.0, it is recognized that the performance of the solution in the On-Time VTP is as good as that in the VTP.

Fig.9 and Fig.10 show the results when BC-pAug89.TL was applied as the observed data set. In Fig.9, the horizontal axis show the time from start of On-Time VTP, the left vertical axis shows the Hurst Parameter and the right vertical axis shows the error rate. From these results, the On-Time VTP achieve low error rate through all period and excellent sensitivity for the dynamic change of traffic property.

Fig.11 and Fig.12 shows the results when BC-pOct89.TL was applied as the observed data set. From those results, as well as the result of BC-pAug89.TL, the performance of the solution in the On-Time VTP was very good.

It is reported that the Hurst Parameter of BC-pAug89.TL is 0.78[1]. However, we find the Hurst Parameter dynamically changes between 0.75 and 0.85. To control a network considering Self-Similarity, it is necessary to know the change in the Hurst Parameter depending on time. By adequately setting the Parameter b, s, a, the On-Time VTP can get the suitable Hurst Parameter in time period which the network control system requires. Consequently, the On-Time VTP can be adopted to more efficient network control.

Then, we try to apply the On-Time VTP to the simple network control, in order to show effectiveness of the network control with Self-Similarity. We set up the simulation environment as illustrated in Fig.13. The shaper adap-

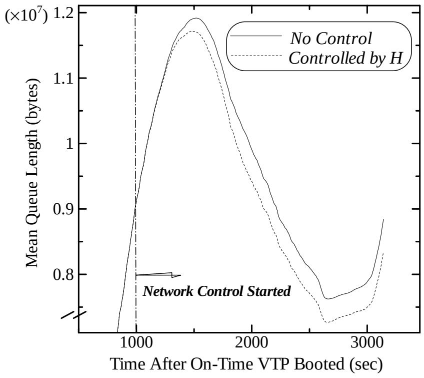

Figure 15: Simple simulation result

tively allocates the output bandwidth based on the Hurst Parameter as shown in Fig.13.

We assume $a = b = 0.1$ in Fig.14. If the Hurst Parameter is 0.5, the bandwidth shaper allocates 90% bandwidth of the output rate. On the other hands, if the Hurst Parameter is 1.0, the shaper allocates 110% bandwidth of the output rate. Figure.15 illustrates the simulation result.

By allocating the output bandwidth considering the Self-Similarity, we can decrease the mean queue length after the On-Time VTP started. From this result, the effectiveness of the network control with Hurst Parameter could be confirmed at least.

6 Conclusions

In this paper, first, we proposed the high-speed calculation method of the Hurst Parameter " On-Time VTP" which aims at being attached in the network control system. Next, we have shown that the On-Time VTP could obtain the solution at much higher speed than the VTP. Moreover, we evaluated the several performance based on real traffic, to verify the effectiveness of the On-Time VTP. As results, we confirmed that the On-Time VTP achieved low error rate through all period and excellent sensitivity for the dynamic change of traffic property. Finally, we adopted On-Time VTP to the network control for evaluation. Then, we showed the effectiveness of the network control considering the Self-Similarity by means of the Hurst Parameter which changes dynamically. In the future, first, we will evaluate the performance by using more measurement data sets and find the suitable parameters of the On-Time VTP to build it in various network control systems. Next, we will make the performance of some network controls such as bandwidth management control, etc. improve by implementing the On-Time VTP mechanism to the control.

References

[1] E.Leland, Murrad S.Taqque, W.Willinger, and Daniel V.Willson "On the Self-Similar Nature of Ethernet Traffic(Extended Version)" *IEEE/ACM Transactions on Networking, Vol.2, No.1, pp.1-15, February 1994.*

[2] W.Willinger, S.Devadhar, A.T.Heybey, R.Sherman, M.Sullivan,andJhon R. Vollaro "Measuring ATM Traffic Cell-by-Cell: Experence and Preliminary Findings from BAGNet" *PMCCN'97 pp.91-110, November 1997.*

[3] M. E. Crovella and A.Bestavros "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes" *IEEE/ACM Transactions on Networking, Vol.5, No.6, pp.835-846, December 1997.*

[4] V.Paxson, S.Floyd "Wide Area Traffic: The Failure of Poisson Modeling" *IEEE/ACM Transactions on Networking, Vol.3, No.3, pp.226-244, June 1995.*

[5] H.Doi, T.Hagiwara, H.Tode and H.Ikeda "Properties of the Multiplexed Self-Similar Traffic on ATM Network" *IEEE APCC/ICCS'98, pp.444-448, November 1998.*

[6] A. Erramilli, O.Narayan and W.Willinger "Experimental Queuing Analysis with Long-range Dependent Packet Traffic." *IEEE/ACM Transactions on Networking, Vol.4, No.2, pp.209-223, April 1996.*

[7] W.Stallings "HIGH-SPEED NETWORKS TCP/IP AND ATM DESIGN PRINCIPLES" *PRENTICE HALL 1998*

[8] Patrick R. Morin and J.Neilson "The Impact of Self-Similarity on Network Performance Analysis *Carleton University Computer Science 95.495, December 4, 1995.*

[9] Z.Sahinoglu and S.Tekinay "On Multimedia Networ ks: Self-Similar Traffic and Network Performance"*IEEE Communication Magazine January 1999, pp.48-52.*

[10] K.Park, G.Kim and M.Crovella "On the Effect of Traffic Self-Similarity on Network Performance" *Proceedings of the 1997 SPIE International Conference on Performance and Control of Network Systems, November, 1997.*

[11] M.Imanishi, T.Hagiwara, H.Doi, H.Tode and H.Ikeda "A Study of Bandwidth Estimation for Self-Similar Traffic." (in Japanese) *Proceedings of the 1999 IEICE General Conference, B-6-21, pp.21, March, 1999.*

[12] H.F. Zhung, Y.T.Shu, and O.Yang "Estimation of Hurst Parameter by Variance-Time Plots" *Proceedings of the IEEE Pacrim 1997, pp.883-886, August 1997.*

[13] K.Park, G.Kim, and M.Crovella On the relationship between file size, transport protocols, and self-similar network traffic *Proceedings of the ICNP, pp.171-180, October, 1996.*

[14] T.Hagiwara, H.Doi, H.Tode and H.Ikeda "The High-Speed Calculation Method of Hurst Parameter" (in Japanese) *Proceedings of the 1999 IEICE General Conference, B-7-20, pp.60. September, 1999.*