Playout Control for Streaming Applications by Statistical Delay Analysis

Kouhei Fujimoto † **Shingo Ata** ‡ **Masayuki Murata** †

†Graduate School of Engineering Science, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka, 560–8531, Japan
Phone +81–6–6850–6616, Fax +81–6–6850–6589

E-mail {k-fujimo, murata}@ics.es.osaka-u.ac.jp
‡Faculty of Engineering, Osaka City University
3–3–138 Sugimoto-cho, Sumiyoshi, Osaka 558–8585, Japan
Phone, Fax +81–6–6605–2191
E-mail ata@info.eng.osaka-cu.ac.jp

Abstract—*Abstract*- **A packet transmission delay is an important quality characteristic for various applications including real-time and data applications. In particular, it is necessary to investigate not only a whole distribution of the packet transmission delay, but also the tail part of the distribution, in order to detect the packet loss. In this paper, we analyze the characteristics of the tail part of packet delay distributions by statistical analytic approach. Our analytic results show that the Pareto distribution is most appropriate in 95–99.9% region of the cumulative distribution of packet transmission delays. Based on our statistical analysis, we next propose an adaptive playout control algorithm, which is suitable to real–time applications. Numerical examples show that our algorithm provides the stable packet loss ratio independently on traffic fluctuations.**

I. Introduction

The Internet is now widely deployed and the users can easily get the global accessibility from their home terminals. One of the main reasons for the prevalence of the Internet is in its routing mechanism. Routing of the Internet has two key features; flexibility and scalability. The Internet provides the dynamic routing based on the exchange of the routing information among routers. For example, when a network link becomes down because of some troubles, an alternative route will be prepared automatically. Second, the packet processing at the routers is simple (e.g., FIFO) to reduce the overhead of packet forwarding at the router.

From the users' point of view, on the other hand, the packet transmission delay is an important metric since it directly affects the end-to-end performance. One example can be found in the real-time application using RTP (Realtime Transport Protocol) [1]; a popular protocol for real-time applications in recent years. RTP uses RTCP (Real Time Control Protocol) to control the transmission rate. In RTCP, the sender maintains the transmission delay of packets based on RTT values to control the packet transfer rate. To keep the preferable performance in RTP-based applications, an accurate estimation of the packet transmission delay is essential. However, RTT estimation is insufficient in some situations. In real-time voice communications, for example, it is desirable to separately measure transmission delays of both downstream (sender to receiver) and upstream (receiver to sender) routes because many of the Internet routes are asymmetric [2]. From these reasons, it is necessary to investigate not only the characteristic of RTT but also that of one-way transmission delays in order to develop an accurate delay estimation method.

However, the Internet is lack of performance guarantees to realize the flexibility and scalability; the dynamic routing of the Internet makes it impossible for the end-users to select the appropriate route for satisfying the users' quality of service (QoS). Furthermore, due to a simple packet processing at routers, it is difficult to predict the transmission delay of the packet. In this paper, we show the accurate packet transmission delay estimation based on the statistical analytic approach.

The studies about the characteristics of the end-to-end packet transmission delay have been made in some literatures [3], [4], but most of those studies have focused on the average characteristics and the entire distributions only. If we want to detect the packet loss, the tail distribution is more important than the entire distribution. For example, in UDP based real-time applications, control of the playout time should be accurate to provide the high-quality real-time service. Here, the playout time is a time when the application client actually begins to play the packet. In the playout control, the client application changes its buffering time, which directly affects the communication quality of the application. While the playout is effective to absorption of the delay variation, too short playout time leads to the fact that the client treats packets to be lost even if those packets eventually arrive. Of course, large playout times may introduce an unacceptable delay that the client user cannot be tolerant. More difficulty exists in determining the playout time. The packet transmission delay between the server and client is changed according to time in the Internet environment. The adequate playout time is heavily dependent on variations of packet transmission delays; i.e., the delay distribution and its time-dependent behavior are also important in determining the playout time.

Keeping those facts in mind, we analyze the characteristics of the packet transmission delays. We first measure the distribution of the one-way transmission delay as well as the round-trip delay, and determine the suitable distribution function through a statistical analytic approach. We next apply the distribution function to estimate the playout time mainly for real-time applications. In an actual situation, some user prefer the real-time reproduction of the media even if the packet loss becomes high, and another user may want high quality at the expense of the large delay. By taking account of it, we propose a new playout control method which ensures the QoS of real-time application according to user's willingness while minimizing the overhead of playout time.

The paper is organized as follows. We first show a brief summary of the characteristics of the packet transmission delay and our measurement framework in Section 2. In Section 3, we explain our analytic approach to estimate parameters of distribution functions and select the most appropriate distribution. We next show the result of analysis in Section 4. In Section 5, we propose a new playout control method based on the results in Section 4, and show the effectiveness of our proposals. Finally, we summarize our work and describe our future research topic in Section 6.

II. METHODS OF PACKET TRANSMISSION DELAY MEASUREMENTS

In this section, we show a brief summary of our measurement method. We measured two types of the packet transmission delay; the round-trip transmission delay and the one-way transmission delay. We first show the outline of the measurement approach, and we next describe our measurement environments.

A. Measurements of the Round Trip Time

There are several tools to measure the RTT. See [5] and references therein. We adopted `pchar` [6] for RTT measurements. `Pchar` (an updated version of `pathchar` [7]) was developed to measure the bandwidth of intermediate links between two end hosts. `Pchar` uses the ICMP (Internet Control Message Protocol) Time Exceeded message to measure the RTT. More specifically, `pchar` utilizes the TTL (Time To Live) field in the IP packet. By protocol specification, the router decreases the value of TTL by one before the packet forwarding. If the value of TTL becomes zero, the router sends the ICMP Time Exceeded packet back to the sender. Thus, `pchar` intentionally sets the value of TTL to a smaller value to indicate the number of hops the packet can traverse. After the sender receives the ICMP Time Exceeded packet, the sender can obtain the RTT which is the duration between when the host sends the packet and when it receives the ICMP packet. The advantage of using ICMP messages is that it is not necessary to deploy any other hosts to measure the RTT. In addition, `pchar` provides events of routing changes and the packet loss ratio. Those are the reasons why we adopted `pchar`.

B. Measurements of the One-way Delay

To measure the one-way delay, we developed the server-client based tool, in which the sender host records the current time into the packet before sending. When the packet arrives at the receiver host, the delay is calculated using the receiver's clock. For this, time clocks of the sender and the receiver should be synchronized. However, the synchronization among distributed hosts in the Internet is difficult and a still open issue [8], [9]. To solve this problem, we use GPS (Global Positioning System) for time synchronization. We measured the one-way delay by considering the following two different types of real-time applications.

- **Sporadic Media:** The voice conversation is classified into this type of applications. It has two periods; the talk-spurt and silent period. The sender transmits the sequence of packets with some intervals during the talk spurt. In the silent period, on the other hand, no packets are generated.
- **Continuous Media:** Data packets are periodically sent by the sender host. The Internet radio and the live event concert are categorized into the continuous media.

C. Measurement Methodology

In our experimental setting, the measurement host is connected to ISP (Internet Service Provider) via 28.8 Kbps telephone line, since we suppose the case that customers use the streaming based real-time application at their home terminals. We measured RTTs to some famous WWW servers in Japan in January 2000. We next measured one-way delays between two hosts which are connected by the 28.8 Kbps modems to different ISPs on July 2000. Throughout our measurements, we also investigate the influences of the following two factors on the determination of suitable distribution functions.

- **Effects of the Time of Day:** It is known that the Internet traffic pattern repeats every day [10]. Thus, it is important to investigate the patterns of the suitable distribution function caused by the effects of "time of day".
- **Effect of the Timescale:** If the timescale for parameter estimation is too short, it may mislead to the wrong estimation. Thus, it is essential to investigate the effect of timescale for the determination of the suitable distribution function.

III. MODELING THE PACKET TRANSMISSION DELAY

In this section, we apply the statistical analysis methods to the measurement data following the method described in [11] where the authors analyzed characteristics of `telnet` and `ftp` traffic. In what follows, we summarize our statistical method.

A. Distribution Functions

We selected four distribution functions as candidates to adequately represent delay distributions. The normal and exponential distributions are given by

$$F(x) = \int_0^x \frac{1}{\sqrt{2\pi}\sigma} exp\left[\frac{-(y-\zeta)^2}{2\sigma^2}\right] dy, \qquad (1)$$

and

$$F(x) = 1 - \exp(-\frac{x}{\beta}), \quad \beta > 0, \qquad (2)$$

respectively. The lognormal distribution is the function, of which variable is the logarithmic variable of the normal distribution, i.e.,

$$F(x) = \int_0^x \frac{1}{\sqrt{2\pi}\sigma y} exp\left[\frac{-(\log y-\zeta)^2}{2\sigma^2}\right] dy. \qquad (3)$$

The Pareto distribution is widely known to be able to represent a self-similarity [12], [13], which is given by

$$F(x) = 1 - \left(\frac{k}{x}\right)^{\alpha}, \quad x \geq k. \qquad (4)$$

B. Parameter Estimation

In order to detect the packet loss from the distribution of packet transmission delays, the coincidence at the tail part of distributions is important, even if the measured data are far from the model distribution function in the other part of the entire distribution. To fit the distribution function accurately, we estimate parameters by only the tail part (e.g., 90–99.9%) of collected delays. For parameter estimations of each distribution function, we use the maximum-likelihood-estimator (MLE) method. Parameters of the exponential and normal distributions can be estimated by calculating the mean and variance of measured delays. In the lognormal distribution, two parameters (ζ, σ) are calculated from

$$\widehat{\zeta} = \frac{1}{n}\sum_{i=1}^{n}\log x_i, \tag{5}$$

$$\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\log x_i - \bar{x}\right)^2, \tag{6}$$

where n is the number of delays. Parameters of the Pareto distribution are obtained from [14];

$$\widehat{k} = \min_i x_i, \tag{7}$$

$$\widehat{\alpha} = n\left[\sum_{i=1}^{n}\log\left(\frac{x_i}{\widehat{k}}\right)\right]^{-1}. \tag{8}$$

C. Determination Method of Adequate Distribution

We determine the most appropriate probability distribution function by χ^2-test. Note again that the coincidence in the tail part of the distribution is most important. Because a typical application of our analysis may be found in the playout control for streaming type applications, estimation of the value around the target point (e.g., 99%, 99.9%) of the delay distribution should be accurate, which directly affects the packet loss ratio in streaming applications and the reproduction quality of real-time applications. From this reason, we evaluate the coincidence between the candidate functions and measured delays on 95–99.9% region of the cumulative distribution by the χ^2-test.

Due to space limitations, we do not show the process of the χ^2-test, and see [15] for a detailed description. In the χ^2-test, the distribution having the smaller value of $\widehat{\lambda}^2$ is more appropriate to represent the measured data. Consequently, we determine that the appropriate model distribution is the candidate distribution which has the smallest $\widehat{\lambda}^2$.

IV. Analytic Results

In this section, we show results of our statistical analysis described in the previous section.

A. Essential Results and Effects of Time of Day

We summarize results of χ^2-test in Table I. The first and second columns of Table I show the type of delay (RTT or One-way) and measured time, respectively. Values of $\widehat{\lambda}^2$ are

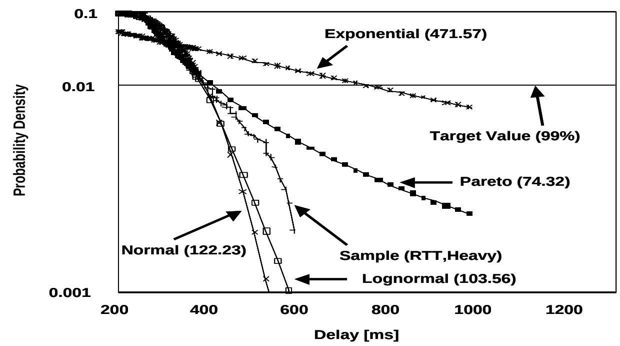


Fig. 1. Comparisons among Sample and Candidate Functions (RTT; 11 PM Hours)

shown in columns 3 through 6. The smallest value of $\widehat{\lambda}^2$ among four distributions is shown in bold. As an example, Figure 1 compares the distribution of the measured RTTs with candidate probability functions in busy hours (corresponding to the second row in Table I). We set the target value to 99% of the cumulative distribution. The distribution labeled by "Sample" is the the tail part (90–99.9%) of the cumulative density distribution of measured RTTs. We next show the cumulative distribution of RTT values during non-busy hours in Figure 2 (the eighth row of the table). It also shows the tail part of the measured RTTs' distribution and candidate probability functions.

We can observe from Table I that $\widehat{\lambda}^2$ of the Pareto distribution is always smallest in all experiments, i.e., the Pareto distribution is most suitable to estimate he 99% value of cumulative distribution in busy hours (e.g., 11 PM[1]) and standard hours (e.g., 2 PM). It is applicable to both RTTs and one-way delays. It coincides the past researches, which showed that the distribution of packet delays is heavy-tailed as the network becomes congested [11].

To illustrate the importance of examining the tail part of the distribution, we next present the characteristics of the distribution in non-busy hours. For this purpose, the χ^2-test is applied to the entire cumulative distribution. Table II shows the result. Comparing with Table I, the model determination method picked up different distributions (normal or lognormal distribution), which were not observed when examining only the tail part of distributions.

Result of Table I has another advantage; as we will describe in the next section, we will apply the statistical results presented in this section to on–line estimation of the delay, which is necessary in playout control. Thus, we want a light-weight estimation method for the delay distribution. Since we found that the Pareto distribution is most appropriate regardless of the "time of day", it is not necessary to examine the χ^2-test for each measurement, and we only have to determine the parameters of the Pareto distribution. If the appropriate model is varied according to the "time of day", we need to examine the χ^2-test for each playout controls. However, the computational overhead of χ^2-test is not small, and it is not adequate for real-time applications.

[1] It is because NTT (one of largest carriers in Japan) offers the service with unlimited accesses at a fixed charge from 11 PM to 8 AM.

TABLE I

RESULTS ON MODEL DETERMINATION (TAIL-PART OF DELAY DISTRIBUTIONS)

Measurement		Result of χ^2-test			
Delay Type	Hour	Nor.	Exp.	Lognor.	Pareto
RTT	10 PM	332.17	2371.91	266.60	**79.75**
RTT	11 PM	122.22	471.56	103.56	**74.32**
RTT	0 AM	156.09	670.34	128.86	**58.45**
RTT	1 AM	157.21	2189.33	139.47	**49.81**
RTT	2 AM	362.24	1691.48	242.74	**115.28**
RTT	7 AM	292.30	3598.50	240.55	**124.03**
RTT	10 AM	169.64	970.60	360.29	**80.57**
RTT	2 PM	147.02	599.37	250.51	**56.25**
RTT	7 PM	194.33	584.95	257.05	**55.63**
One-way	9 PM	83.82	602.56	71.96	**19.56**
One-way	11 PM	53.86	470.90	49.67	**30.10**
One-way	1 AM	55.06	426.46	49.99	**24.01**
One-way	5 AM	94.45	500.91	85.77	**25.16**
One-way	9 AM	107.76	754.09	98.74	**45.33**
One-way	12 PM	108.66	1218.95	101.09	**30.61**
One-way	3 PM	109.07	336.49	85.41	**21.21**

TABLE II

RESULTS ON MODEL DETERMINATION (ENTIRE DELAY DISTRIBUTIONS)

Measurement		Result of χ^2-test			
Delay Type	Hour	Nor.	Exp.	Lognor.	Pareto
RTT	11 PM	173.59	830.91	126.45	**100.22**
RTT	1 AM	164.39	1136.62	**130.49**	130.64
RTT	7 AM	154.59	1780.39	**97.49**	189.54
RTT	10 AM	**21.09**	49.27	32.16	36.873
RTT	2 PM	**22.07**	46.27	26.75	34.51

B. Effects of Timescale

We next examine the effects of the timescale by changing the number of samples for the parameter estimation. Figures 4(a) and 4(b) shows the degree of differences against the number of measured data for RTT and one-way delays, respectively. We calculate the difference between the 99% values of the Pareto distribution and those of the cumulative distribution of collected samples. As shown in the figure, the difference is remarkable as the number of samples are less than 500. On the other hand, we cannot observe critical changes when the number of measurement data are equal or more than 500. Since our objective is to perform on-line estimation of the delay distribution, it is preferable that the number of sample is as small as possible, then the parameter can be estimated faster with the less number of samples. From the results, we can conclude that the required number of measurements should be equal to or more than 500, in which $500 \times (99.9\% - 90\%) \simeq 50$ samples are at least necessary for the accurate parameter estimation of the Pareto distribution. We will evaluate the required number of samples for quick and still accurate parameter estimation in the next section.

V. PLAYOUT TIME ESTIMATION METHOD BASED ON STATISTICAL ANALYSIS

In this section, we propose a new playout control algorithm in which playout time is determined based on our statistical analysis. Then, we evaluate our playout control algorithm by the trace-driven simulation, and we investigate effectiveness of the proposed algorithm.

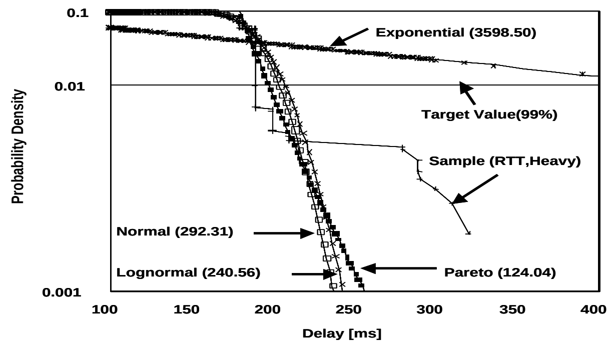


Fig. 2. Comparisons among Sample and Candidate Functions (RTT, 2 PM Hours)

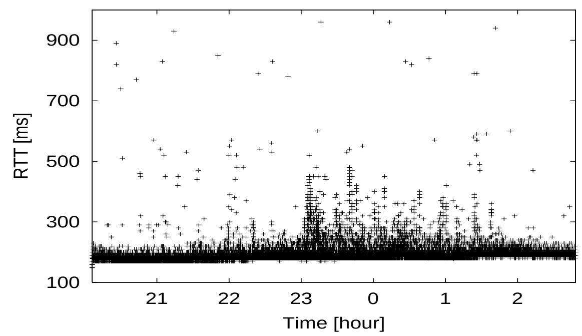


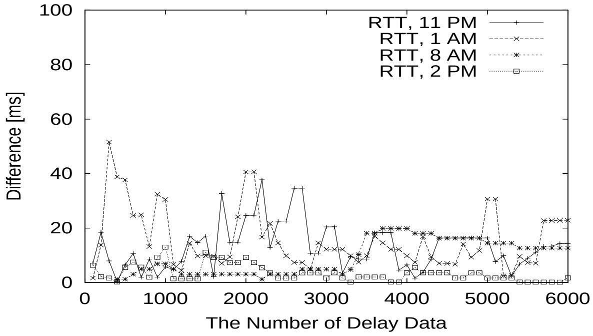
Fig. 3. Variations in RTTs

A. Proposed Algorithm

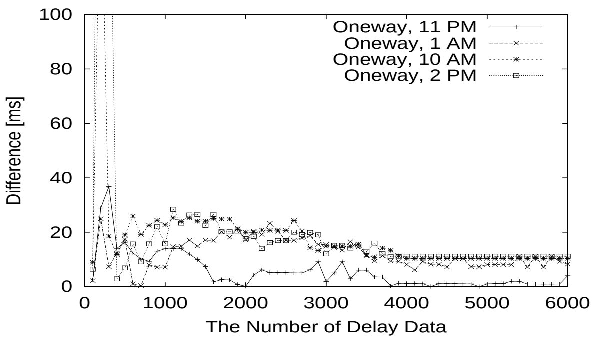
To provide a high-quality communication in real-time applications, the packet loss ratio should be kept small. Because packets arriving after the playout time is not meaningful, a choice of playout time should be performed carefully. In addition, the playout control should provide some means to determine the quality level of "real-time" transmission of the media that the user is acceptable. Very large playout time can lead to less packet loss, but it degrades the "real-time" reproduction of the media. The main goal of our algorithm is to minimize the playout time while keeping the reproduction quality specified by user's requirements. We use the results obtained through the statistical analysis presented in the previous section to determine the proper playout time.

More specifically, our playout algorithm records the history of one-way delays of packets. On each packet arrival, parameters of the Pareto cumulative density function $F(x)$ is updated to estimate the playout time p_i from the equation $F(p_i) = X$ where X is the target value. In this paper, we consider 95, 99, and 99.9% as the target value X. For example, if we choose $X = 95\%$, our algorithm tries to minimize the playout time while keeping the packet loss to be 5%. Of course, if the packet loss within the network exceeds 5%, our method has no means to keep the packet loss to be 5%. In what follows, we will assume that the packet loss within the network does not exceed the target value.

In what follows, we will provide the trace-driven simulation

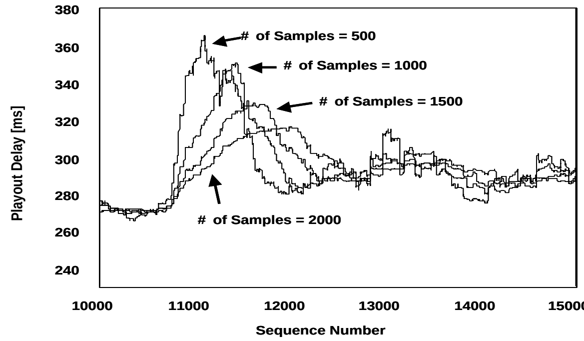


(a) RTT

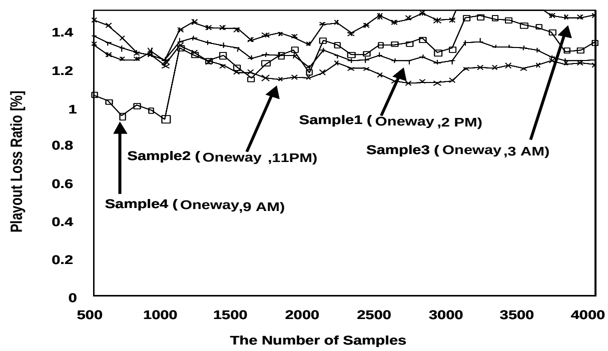


(b) One-way

Fig. 4. The Difference between The Value of The Pareto Distribution and The Measured Data at The Target Value (99%).



(a) Variations of Playout Times



(b) Playout Loss Ratio dependent on the Number of Measured Samples

Fig. 5. Effect of the Number of Samples in Playout Control

results. A collected set of one-way delays of packets is used in simulation. The packet size was set to be 160 bytes, and an interval of packet emissions fixed at 80 msec. We then estimate the playout time p_i of the packet i according to the algorithm presented above. In simulation, we check whether the next packet arrives within the estimated playout time or not, and if the packet does not arrive, it is treated as packet loss. That is, we do not take account of the packet loss within the network in simulation.

B. Parameter Setting

In our algorithm, the number of measurements for the parameter estimation becomes a dominant factor. The accuracy of parameter estimation can be improved by increasing the number of measurement data. However, the larger number of samples inhibits to follow the dynamic changes of the network condition, and the playout control cannot follow the drastic variation of one-way delays.

We have described in Subsection 4.2 that the number of samples should be more than 500 for an accurate parameter estimation. We now demonstrate the influence of the number of samples on playout controls by changing the number of samples. The results of experiments are shown in Figure 5. Figure 5(a) shows the variation of playout time in two cases; 500 and 2,000. It shows that the smaller the number of samples is, the more quickly the playout time changes. Then, the problem is how accurate the playout control can estimate the parameters of the Pareto distribution, and determine the playout time to follow the target packet loss rate. The results are shown in Figure 5(b) where the packet loss rate is plotted against the number of samples. In the figure, we plot two cases of the busiest (11 PM) and non-busiest hours (2 PM). In obtaining the figure, we set the target packet loss ratio to be 1%. In the figure, we cannot observe a significant improvements even if the number of samples to be large. It is because the improvement of accuracy with the increased number of samples in parameter estimation is canceled out by the degradation of adaptability to delay variations. Based on above results, the number of samples can be set to be 500 for parameter estimation of playout controls in our algorithm.

C. Performance Comparisons

For comparison purpose, we also examined two algorithms which have been proposed in [16]. Note here that we refer to our proposed algorithm as **Algorithm 1** throughout in this section.

TABLE III

COMPARISON OF PLR AND MEAN PLAYOUT TIME

Algorithm	Target Value	PLR [%]	Mean Playout Time [ms]
1	95%	5.13	221.17
	99%	1.37	265.12
	99.9%	0.14	855.38
2	-	2.46	237.16
3	-	0.24	392.64

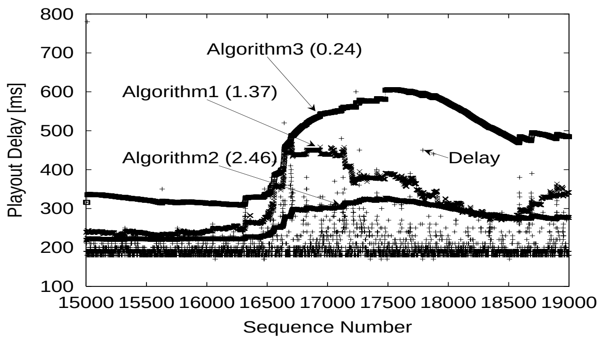


Fig. 6. Playout Time Variations of Algorithm 1 to 3

In **Algorithm 2**, the playout time is determined from the approximated values of the mean $\widehat{d}_i$ and the variance $\widehat{v}_i$ of one-way delays, which is given by

$$p_i \quad = \quad t_i + \widehat{d}_i + 4\widehat{v}_i. \tag{9}$$

That is, the playout time is decided without a knowledge on the delay distribution. **Algorithm 3** is a modified version of Algorithm 2. It uses the weighted mean of $\widehat{d}'_i$s as

$$\widehat{d}'_i = \begin{cases} \beta\widehat{d}'_{i-1} + (1-\beta)n_i & \text{if } n_i > \widehat{d}'_{i-1}, \\ \alpha\widehat{d}'_{i-1} + (1-\alpha)n_i & \text{otherwise,} \end{cases} \tag{10}$$

where α and β are constant value which satisfies $0 < \beta < \alpha < 1$. We set $\alpha = 0.998500$ and $\beta = 0.970000$ according to the suggestion in [16].

Table III compares packet loss ratios (PLRs) and mean values of the playout time in busy hours. In Algorithm 1, we used 95, 99, and 99.9% as the target values. We can observe that there is a clear trade-off between PLR values and the playout time. The advantage of our proposed algorithms is that the value of PLR can be kept close to the desired packet loss ratio $(1 - X)$. Namely, it is easy to control the quality of real-time applications by changing the target PLR. Of course, in Algorithms 2 and 3, the target value of PLR might be controlled by changing the multiplier of $\widehat{v}_i$, which is currently set to be 4. See Eq. (9). However, there is no means to map the multiplier to the value of PLR in those algorithms.

Figure 6 compares the playout time variation among three algorithms in busy hours. The target value X is set at 99% in Algorithm 1. From this figure, we can find that Algorithm 3 has a tendency to overestimate the playout time which results in less packet loss. However, Table III also shows that Algoritm 3 requires too large playout time which is about 50% larger than that of Algorithm 1.

VI. CONCLUDING REMARKS

In this paper, we have measured packet transmission delays and analyzed their characteristics by taking into account the time of day.

From statistically analytic results, we have found that the Pareto distribution is most appropriate as the model of one-way delay distribution, as well as RTT distributions. Moreover, we have proposed a playout control algorithm based on our analysis. Numerical examples have shown that our proposed method can control the playout time in order to satisfy the target packet loss probability. For future research topics, it is necessary consider the update process of the playout time in order to apply our algorithm to actual real-time applications.

ACKNOWLEDGEMENTS

This work was supported in part by Research for the Future Program of Japan Society for the Promotion of Science under the Project "Integrated Network Architecture for Advanced Multimedia Application Systems"(JSPS-RFTF97R16301).

REFERENCES

[1] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," *RFC 1889*, January 1996.

[2] S. Savage, A. Collins, and E. Hoffman, "The end-to-end effects of Internet path selection," in *Proceedings of ACM SIGCOMM '99*, pp. 289–299, September 1999.

[3] J.-C. Bolot, "Characterizing end–to–end packet delay and loss in the Internet," in *Proceedings of ACM SIGCOMM '97*, pp. 289–298, September 1997.

[4] V. Paxson, "End–to–end Internet packet dynamics," in *Proceedings of ACM SIGCOMM '97*, pp. 139–152, September 1997.

[5] "Caida measurement tool taxonomy," `http://www.caida.org/tools/taxonomy/index.xml`.

[6] B. A. Mah, "pchar: A tool for measuring Internet path characteristics," `http://www.ca.sandia.gov/~bmah/Software/pchar/`.

[7] A. B. Downey, "Using pathchar to estimate Internet link characteristics," in *Proceedings of ACM SIGCOMM '99*, pp. 241–250, August 1999.

[8] V. Paxson, "Measurements and analysis of end-to-end Internet dynamics," *Ph.D.Thesis, University of California Berkeley*, April 1997.

[9] S. Kalidindi and M. J. Zekauskas, "Surveyor: An infrastructure for Internet performance measurements," *in Proceedings of INET '99 available at* `http://www.advanced.org/csg-ippm/`, June 1999.

[10] K. Thompson, G. J. Miller, and R. Wilder, "Wide–area Internet traffic patterns and characteristics," *IEEE Network*, pp. 10–23, November 1997.

[11] V. Paxson and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," in *Proceedings of ACM SIGCOMM '94*, pp. 257–268, August 1994.

[12] W. E. Leland, M. S. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic," in *Proceedings of ACM SIGCOMM '93*, vol. 2, pp. 183–193, February 1993.

[13] M. E. Crovella and A. Bestavros, "Self-similarity in World Wide Web; traffic evidence and possible causes," in *Proceedings of ACM SIGMETRICS '96*, pp. 160–169, May 1996.

[14] V. Brazauskas and R. Serfling, "Robust and efficient estimation of the tail index of a one-parameter pareto distribution," *North American Actuarial Journal available at* `http://www.utdallas.edu/~serfling/`, April 2000.

[15] S. Pederson and M. Johnson, "Estimating model discrepency," *Technometrics*, vol. 32, pp. 305–314, August 1990.

[16] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive playout mechanisms for packetized audio applications in wide–area networks," in *Proceedings of IEEE INFOCOM '94*, pp. 680–688, April 1994.