On Modeling Round-Trip Time Dynamics of the Internet using System Identification

Hiroyuki Ohsaki[1], Mitsushige Morita[2], and Masayuki Murata[1]

[1] Cybermedia Center, Osaka University
1-30 Machikaneyama, Toyonaka, Osaka 567-0043, Japan
{oosaki, murata}@cmc.osaka-u.ac.jp

[2] Graduate School of Engineering Science, Osaka University
1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan
m-morita@ics.es.osaka-u.ac.jp

Abstract. Understanding the end-to-end packet delay dynamics of the Internet is of crucial importance since it directly affects the QoS (Quality of Services) of realtime services, and it enables us to design an efficient congestion control mechanism. In this paper, we measure the round-trip time, and build a mathematical model representing its dynamics using system identification. We first measure, as the input and output data for system identification, the packet inter-departure time from a source host and the corresponding round-trip time measured by the source host. ICMP (Internet Control Message Protocol) is utilized to measure the round-trip time for each packet. We next model the network, seen by a specific source host, as a dynamic SISO (Single-Input and Single-Output) system. Using measurement results obtained from three different network configurations, we investigate how accurately the round-trip time dynamics of the Internet can be modeled with the system identification.

1 Introduction

Understanding the end-to-end packet delay dynamics of the Internet is of crucial importance since (1) it directly affects the QoS (Quality of Services) of realtime applications, and (2) it enables us to design an efficient congestion control mechanism for both realtime and non-realtime applications. For non-realtime applications, a delay-based approach for congestion control mechanisms, rather than a loss-based approach as used in TCP (Transmission Control Protocol), has been proposed (e.g., [1, 2]). The main advantage of such a delay-based approach is, if it is properly designed, packet losses can be prevented by anticipating impending congestion from increasing packet delays.

For a long time, queueing theory has been extensively used as a powerful tool to analyze packet-switched networks. In general, the queueing theory assumes stationarity of the network, and allows us to obtain several performance measures such as the average packet delay and the average packet loss probability. However, the stringent limitation of the queuing theory is its difficulty to analyze the *dynamic behavior* of the network. Several measurement-based studies suggest that the end-to-end packet behavior in the Internet is quite dynamic [3–5]. Another approach, being different from the queueing theory, should therefore be taken to investigate the packet delay dynamics of the Internet.

In [6], the authors have proposed a novel approach to model the end-to-end packet delay dynamics of the Internet. The main idea of the approach is treating the network, seen by a specific source and destination pair, as a *black-box*, and modeling the end-to-end packet delay dynamics using *system identification* [7]. The end-to-end packet delay dynamics are modeled as a SISO (Single-Input and Single-Output) system based on the ARX (Auto-Regressive eXogenous) model. The input to the system is a packet inter-departure time from the source host, and the output is a (one-way) end-to-end packet delay variation measured by the destination host.

This paper is a direct extension of [6], and primarily focuses on an applicability of our approach to real networks. However, there is a major difference in the modeling approach. In [6], the network seen by a specific source and destination pair is modeled as a black-box. On the contrary, in this paper, the network seen by a specific source host is modeled; that is, the output is the round-trip time variation instead of the end-to-end packet delay variation. Although modeling the round-trip time dynamics suffers from more measurement noise than modeling the end-to-end packet delay, the modeling approach taken in this paper is easier to implement, so that desirable for practical purposes.

After discussing advantages and disadvantages of several measurement methods for the round-trip time, we present a measurement method using ICMP (Internet Control Message Protocol) to collect the input and output data for determining the coefficients of the ARX model. Since almost all hosts and routers respond to ICMP packets, this method can be used in various network environments. We then collect the input and output data from real networks. Three network configurations are used including both wired and wireless LANs. Using the input and output data obtained, coefficients of the ARX model are determined using the least-square method. We investigate how accurately the ARX model can represent the round-trip time dynamics of the Internet.

This paper is organized as follows. In Section 2, we summarizes related works in recent publications. In Section 3, a black-box approach for modeling the round-trip time dynamics of the Internet using the ARX model is explained. In Section 4, we discuss several measurement methods of the round-trip time, in particular, for collecting the input and output data for system identification. Section 5 shows several measurement and modeling results, and discuss how accurately the ARX model can capture the round-trip time dynamics. In Section 6, we discuss several possible applications of our approach, followed by conclusion of this paper.

2 Related Works

In the literature, there have been several measurement-based studies regarding the end-to-end packet delay [3, 4, 8, 9] and the end-to-end path characteristics [5, 10]. In [3], the authors have examined the end-to-end packet delay and loss behavior in the Internet using small UDP probe packets. In [4], the authors have examined the correlation between packet delay and packet loss experienced by a continuous-media traffic source, based on measurements of per-packet delays and packet loss. In [8], a large number of TCP measurements have been used to discuss two estimation problems: estimation of the retransmission timer (RTO) for a TCP connection, and estimation of the available bandwidth. In [9], the

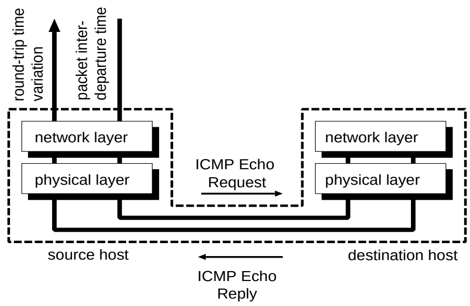


Fig. 1. Modeling round-trip time dynamics as SISO system

authors have presented an approach to characterize loss and delay characteristics of a transmission link based on end-to-end multicast measurements. In [5], the packet dynamics of the Internet have been analyzed based on measurements of about 20,000 TCP data transfers. In [10], the routing behavior of the Internet has been analyzed based on measurements of about 40,000 end-to-end traceroute results. However, those studies are limited to a statistical behavior of the end-to-end packet delays and/or path characteristics. In other words, the dynamics of the packet delay of the Internet, which is the main concern of this paper, has not been investigated.

Aside from analyses of the end-to-end packet delay, another area of measurement-based studies is regarding a black-box modeling of the network traffic [11–15]. In [11], the authors have proposed a traffic model for wide-area TCP traffic by characterizing several distributions of, for example, the packet inter-arrival time and the number of bytes transferred. In [12], the authors have proposed a fast algorithm to construct a CMRP (Circulant Modulated Rate Process) for traffic modeling. In [13], CMRP and ARMA (Auto-Regressive Moving Average) have been discussed as a traffic model. In [14, 15], a measurement-based tool for traffic modeling and queueing analysis has been developed, which uses CMPP (Circulant Modulated Poisson Process) for a traffic model. Those studies are closely related to our black-box modeling approach, but there is a significant difference. Those studies have focused on traffic modeling based only on outputs (i.e., observed amount of traffic). On the contrary, this paper focuses on modeling the round-trip time dynamics based on both inputs (i.e., packet inter-departure time) and outputs (i.e., round-trip time variation). In other words, this paper focuses on how the round-trip time of a packet sent from a source host is affected by its past packet transmission process.

3 Black-Box Modeling and System Identification

As depicted in Fig. 1, the network seen by a specific source host, including underlying protocol layers (e..g, physical, data-link, and network layers), is considered as a black-box. Our goal of this paper is to model a SISO system describing the round-trip time dynamics: i.e., the relation between a packet sending process from the source host and its resulting round-trip time observed at the source host. Effects of other traffic (i.e., packets coming from other hosts) are modeled as *noise*. As the input to the system, we use a *packet inter-departure time* from

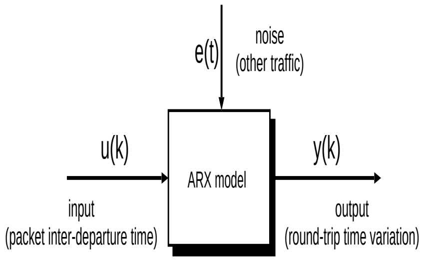


Fig. 2. ARX model for modeling round-trip time dynamics

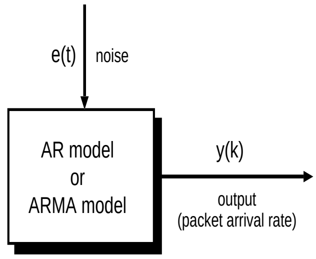


Fig. 3. AR model or ARMA model for modeling network traffic

the source host: i.e., the time interval between two consecutive packet transmissions from the source host. Use of the packet inter-departure time is straightforward since it directly affects the end-to-end packet delay. As the output from the system, we use a *round-trip time variation* measured by the source host: i.e., the difference in two consecutive round-trip times. We choose the round-trip time variation, instead of the round-trip time itself, as the output from the system. This choice is for reducing unstationarity of noise (i.e., effect of other traffic) on the measured round-trip time since the aggregated network traffic at a packet-level time scale is not stationary [16].

In this paper, the ARX model is used and its coefficients are determined using system identification [7]. Figure 2 illustrates a fundamental concept of using the ARX model for modeling the packet delay dynamics. The input to the ARX model is a packet inter-departure time from the source host, and the output from the ARX model is a round-trip time variation measured at the source host. Effects of other traffic (i.e., packets coming from other hosts) are modeled as the noise to the ARX model. Letting $u(k)$ and $y(k)$ be the input and the output data at slot k, the ARX model is defend as

$$A(q)\,y(k) = B(q)\,u(k - n_d) + e(k) \tag{1}$$

where $A(q)$ and $B(q)$ are given by

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{n_a} q^{-n_a}$$
$$B(q) = b_1 + b_2 q^{-1} + \ldots + b_{n_b} q^{-n_b+1}$$

In the above equations, $e(k)$ is unmeasurable disturbance (i.e., noise), and q^{-1} is the delay operator; i.e., $q^{-1}u(k) \equiv u(k-1)$. The numbers n_a and n_b are the orders of polynomials. The number n_d corresponds to delays from the input to the output. For compact notation, ζ is introduced as

$$\zeta = [n_a, n_b, n_d] \tag{2}$$

In our case, $u(k)$ and $y(k)$ correspond to k-th packet inter-departure time and k-th round-trip time variation. All coefficients of the polynomials, a_n and b_n, are parameters of the ARX model, and are to be determined from input and output data using system identification. Refer to [7] for the detail of the ARX model.

Our approach of a black-box modeling using the ARX model is distinctive from other black-box approaches, which model network traffic using the

AR (Auto-Regressive) model or the ARMA (Auto-Regressive Moving Average) model [13, 17, 18]. Figure 3 illustrates a typical usage of the AR model or the ARMA model for modeling network traffic. Comparing Figs. 2 and 3, the ARX model has the input whereas either the AR model or the ARMA model does not. In other words, only the ARX model can represent the dynamics, i.e., how the past input data affects the future output data.

Note that the ARX model has a drawback in modeling the round-trip time dynamics; i.e., the ARX model is a linear time-invariant model, so it cannot rigorously capture non-linearity of the round-trip time dynamics. But it should be noted that the ARX model is applicable in various control engineering problems. This is because non-linear dynamical systems operating around the stable point can be well approximated by a linear system [7]. In Section 5, we will investigate how accurately the round-trip time dynamics can be described by the ARX model.

The system identification problem for the ARX model is formulated as a minimization problem, where the cost function is given by a loss function [7]. Because of space limitation, only the outline is shown in this paper, and interested readers should refer to [7] for more detail.

Let θ be a vector of all coefficients and $\psi(k)$ be a vector of all past n_a outputs and n_b inputs, respectively.

$$\theta = [a_1, \ldots, a_{n_a}, b_1, \ldots b_{n_b}]^T \tag{3}$$

$$\begin{aligned}\psi(k) = [&-y(k-1), \ldots, -y(k-n_a), \\ &u(k-n_d-1), \ldots, u(k-n_d-n_b)]^T\end{aligned} \tag{4}$$

Using Eq. (1), the output from the ARX model $\hat{y}(k|\theta)$ is given by

$$\hat{y}(k|\theta) = \psi^T(k)\,\theta \tag{5}$$

The loss function $V_N(\theta, Z^N)$ is defined as the sum of all squared prediction errors for N input and output data.

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - \hat{y}(k|\theta))^2 \tag{6}$$

where Z^n is the past input and output data defined as

$$Z^N = \{u(1), y(1), \ldots, u(N), y(N)\} \tag{7}$$

The solution $\hat{\theta}_N$ that minimizes the above loss function is easily obtained by the least squares method:

$$\hat{\theta}_N = \left[\sum_{k=1}^{N}\psi(k)\psi^T(k)\right]^{-1}\sum_{k=1}^{N}\psi(k)y(k) \tag{8}$$

4 Measurement Methods

For collecting the input and output data from real networks, it is necessary to send a series of probe packets into the network, and to measure their resulting round-trip times. For sending probe packets, one of the following protocols can be used.

- TCP (Transmission Control Protocol)
- UDP (User Datagram Protocol)
- ICMP (Internet Control Message Protocol)

In what follows, we discuss advantages and disadvantages of these protocols for sending probe packets to collect the input and output data, in particular, for system identification.

TCP has a feedback-based congestion control mechanism, which controls the packet sending process from a source host according to the congestion status of the network. Since it is an ACK-based protocol, it is easy for the source host to measure the round-trip time for each packet. However, TCP is not suitable for sending a probe packet because of the following reasons. First, for system identification purposes, the input data (i.e., the packet inter-departure time) should contain diverse frequencies. So the white noise, which equally contains all frequencies, is the ideal input data for system identification [7]. However, the packet inter-departure process of TCP would have limited frequencies. Second, most of system identification techniques assume an independence between the input and output data. However, because of a feedback-based nature of TCP, the packet inter-departure time is dependent on the past round-trip times, so the independence assumption cannot be satisfied with TCP.

On the contrary, UDP has no feedback-based control. The packet inter-departure time of UDP can be freely controlled. However, UDP is a one-way protocol. The destination host must perform some procedure to measure the round-trip time for each packet at the sender side. One possible way is to use *ICMP Destination Unreachable* message as in the *traceroute* program [19]. When the host receives a UDP packet to an unreachable port, it returns ICMP Destination Unreachable message to the source host. The source host can therefore measure the round-trip time by observing the elapsed time between the UDP packet transmission and the receipt of the corresponding ICMP packet. However, as specified in [20], generation of ICMP Destination Unreachable messages is limited to a low rate. Use of ICMP Destination Unreachable message is therefore not desirable to collect the input and output data for system identification.

ICMP is a protocol to exchange control messages such as routing information and node failures [21]. Since ICMP has no feedback-based control, the inter-departure time of ICMP packets can be freely controlled. Also it is easy to measure the round-trip time at the source host by using *ICMP Echo Request* and *ICMP Echo Reply* messages, as in the *ping* program. Although a part of network devices have a rate limitation for transmitting ICMP Echo messages [22], many network devices respond to ICMP Echo messages. So this method can be used in various network environments.

In this paper, we therefore choose ICMP Echo message as a probe packet. More specifically, the source host sends a series of ICMP Echo Request messages to the destination host, and the destination host returns ICMP Echo Reply messages. We have modified the ping program to dynamically change the packet inter-departure time (originally fixed at one second).

The detailed algorithm is described below.

Sender Algorithm (Source Host):

S1) Send ICMP Echo Request message of 1,500 bytes including IP and ICMP headers. The payload of the ICMP packet holds the timestamp of the packet transmission.

S2) Randomly choose the packet inter-departure time from the exponential distribution in order to schedule the next ICMP packet transmission.
S3) Go to S1.

Receiver Algorithm (Source Host):

R1) Wait for the receipt of ICMP Echo Reply message.
R2) Extract the timestamp from the payload.
R3) Calculate the round-trip time from the current time.
R4) Calculate the round-trip time variation from the previous round-trip time.
R5) Go to R1.

The destination host copies the payload of the received ICMP Echo Request message to the returning ICMP Echo Reply message. Thus, the ICMP Echo Reply packet contains the timestamp placed by the source host at its transmission time. This enables precise measurement of the round-trip time at the source host.

5 Modeling from Measured Data

Network Configurations

We have measured three sets of input and output data from the following three network configurations.

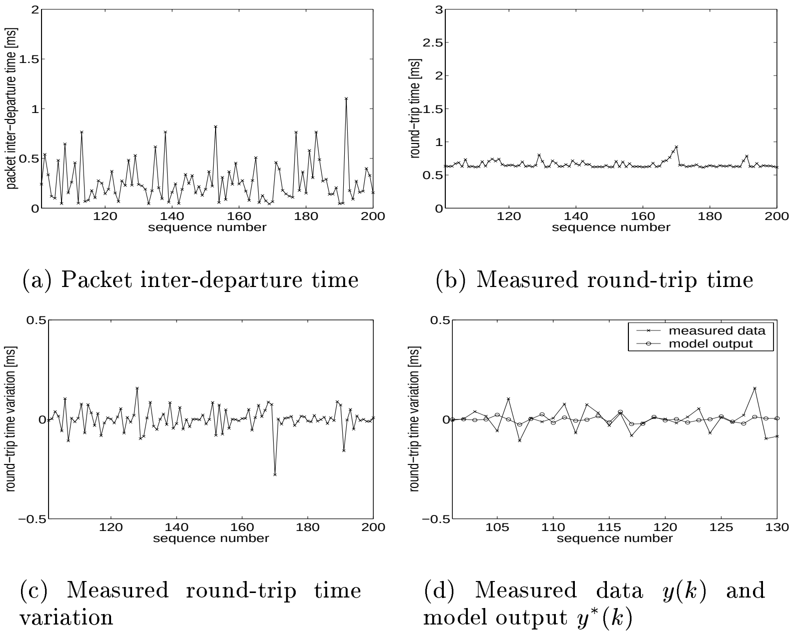
N1) 100 Mbps LAN without background traffic
N2) 100 Mbps LAN with background traffic
N3) 11 Mbps wireless LAN and 100 Mbps LAN

In the network configuration **N1**, both the source and destination hosts are directly connected to a single 100 Mbps switch. Because of a direct connection, there exists no background traffic, and the output data (i.e., the round-trip time variation) suffers from little observation noise. This network configuration enables us to investigate how accurately the round-trip time dynamics can be modeled in a high-speed and non-congested network.

The network configuration **N2** is a 100 Mbps LAN, which consists of five 100 Mbps switches connected in serial. The network configuration **N2** is a private LAN in our laboratory, where about 50 client computers and 10 server computers are connected. There are five switches between the source and destination hosts. Since intermediate switches process traffic from other computers, the round-trip time measured at the source host might be affected by existence of the background traffic. This network configuration is for investigating how the background traffic deteriorates the accuracy of the ARX model.

In the network configuration **N3**, the destination host is equipped with a 11 Mbps wireless LAN interface. The base station is connected to the network configuration **N2**. There are five 100 Mbps switches between the source host and the base station. In this case, the wireless LAN, which is much slower than 100 Mbps LAN, is the bottleneck. The round-trip time is expected to be significantly larger than other network configurations.

In each network configuration, we have collected both the packet inter-departure time $u(k)$ and the round-trip time variation $y(k)$ using the approach



(a) Packet inter-departure time

(b) Measured round-trip time

(c) Measured round-trip time variation

(d) Measured data $y(k)$ and model output $y^*(k)$

Fig. 4. Results in network configuration N1 (100 Mbps LAN w/o Background Traffic)

described in Section 4. The source host sent 10,000 probe packets with an exponentially distributed inter-departure time. Note that lost packets are not included in the measured input and output data. Of all input and output data collected, we use the input and output data of 100 packet samples for coefficients determination and model validation of the ARX model. In what follows, we discuss how accurately the round-trip time dynamics can be modeled by the ARX model.

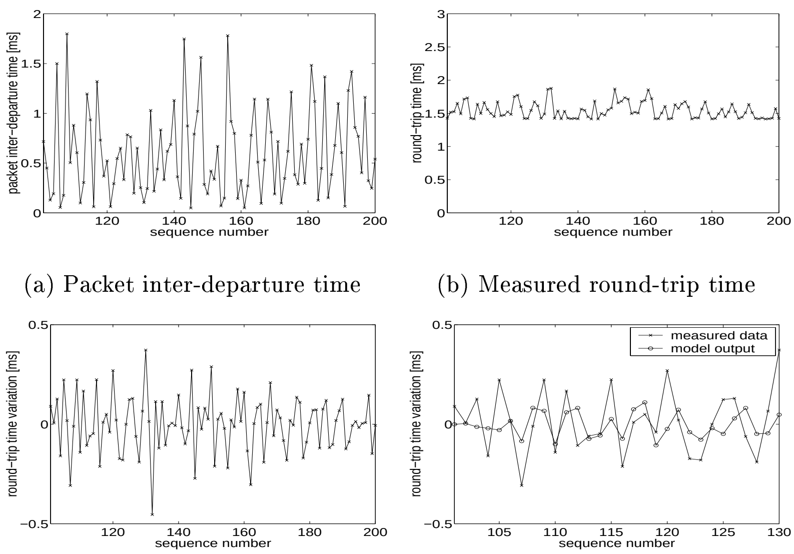
Network Configuration N1

In the network configuration **N1**, the mean packet inter-departure time has been set to 0.2 ms, resulting in the average packet transmission rate of 43.2 Mbps and the average round-trip time of 0.8 ms. Shown in Fig. 4 are results in the network configuration **N1** for $\zeta = [8, 8, 1]$. This figure shows: (a) the packet inter-departure time $u(k)$, (b) the measured round-trip time, (c) the measured round-trip time variation $y(k)$, and (d) comparison between the measured output data and the model output. More specifically, the "measured output data" is the measured round-trip time variation $y(k)$, and the "model output" is the simulated output from the ARX model, which is defined as

$$y^*(k|\theta) = \psi^{*T}(k|\theta)\,\theta \tag{9}$$

where

$$\begin{aligned}\psi^*(k|\theta) = [&-y^*(k-1|\theta), \ldots, -y^*(k-n_a|\theta),\\ &u(k-n_d-1), \ldots, u(k-n_d-n_b)]\end{aligned} \tag{10}$$



(a) Packet inter-departure time (b) Measured round-trip time

(c) Measured round-trip time variation (d) Measured data $y(k)$ and model output $y^*(k)$

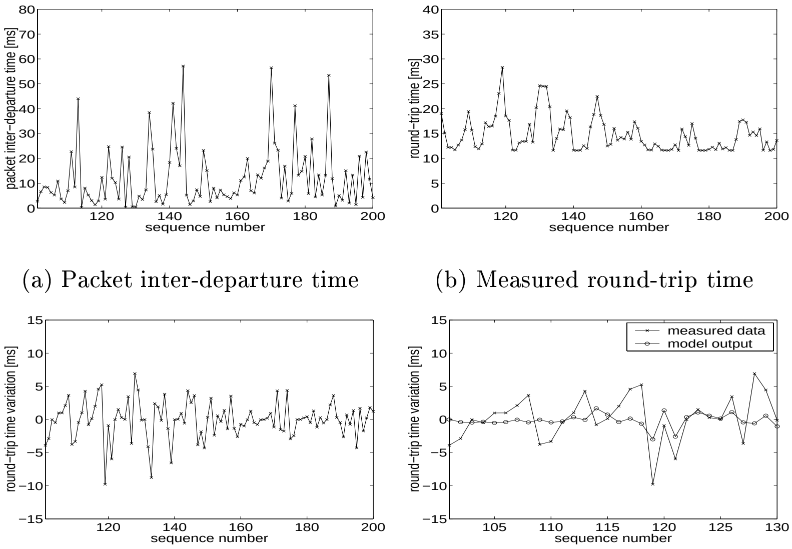
Fig. 5. Results in network configuration N2 (100 Mbps LAN w/ Background Traffic)

Note the difference between $\hat{y}(k|\theta)$ and $y^*(k|\theta)$; i.e., $\hat{y}(k)$ is a 1-step ahead prediction from the measured inputs and outputs, whereas $y^*(k|\theta)$ is a simulated output only from the measured inputs assuming zero noise. There are several techniques for checking accuracy of the ARX model obtained by system identification [7]. Comparing the measured output data and the model output is one of the most intuitive approaches.

Figure 4(c) shows that the amplitude of the round-trip time variation is very small, whereas the packet inter-departure time dynamically changes. This is because there is no background traffic between the source and destination hosts. A slight change in the round-trip time would be caused by the processing delay variation at the host and/or by a timer granularity of the operating system, since the network is not a bottleneck in the network configuration **N1**. Figure 4(d) indicates that the ARX model cannot capture the round-trip time dynamics in the network configuration **N1**. Namely, the model output $y^*(k)$ is almost unchanged, although the measured round-trip time changes. This is caused by the weak correlation between the packet inter-departure time and the measured round-trip time; that is, in the network configuration **N1**, the round-trip time is almost independent of the packet inter-departure time.

Network Configuration N2

Figure 5 shows results in the network configuration **N2** for $\zeta = [8, 8, 1]$. In this case, the mean packet inter-departure time has been set to 0.6 ms, resulting the average packet transmission rate of 18.0 Mbps and the average round-trip time of 1.8 ms. Figure 5(c) shows that the amplitude of the round-trip time variation



(a) Packet inter-departure time

(b) Measured round-trip time

(c) Measured round-trip time variation

(d) Measured data $y(k)$ and model output $y^*(k)$

Fig. 6. Results in network configuration N3 (11 Mbps Wireless LAN + 100 Mbps LAN)

is larger than that of the network configuration **N1**. The main reason for such a large amplitude would be the effect of the background traffic; that is, the round-trip time tends to become large when the network is congested. It can be found that the model output $y^*(k|\theta)$ and the measured output $y(k)$ roughly coincide but slightly differ. This is because the measured round-trip time variation is disturbed by other traffic, which is unknown so that not included in the model output $y^*(k)$.

Network Configuration N3

Results in the network configuration **N3** for $\zeta = [8, 8, 1]$ are shown in Fig. 6. In this case, the mean packet inter-departure time has been set to 12.0 ms, resulting the average packet transmission rate of 967 Kbps and the average round-trip time of 16.7 ms. Figure 6(c) shows that the amplitude of the round-trip time variation is much larger (about 10 ms) than the previous cases, **N1** and **N2**. Figure 6(d) indicates that the round-trip time dynamics is not correctly modeled by the ARX model. It is probably because the transmission delay at the wireless link is significantly changed, resulting in a large measurement noise. From these observations, we conclude that the round-trip time dynamics can be modeled by the ARX model when the network is moderately congested.

Choice of Model Orders and Number of Samples

In the above results, the orders and the delay of the ARX model is fixed at $\zeta = [8, 8, 1]$. In general, the accuracy of the ARX model is dependent on the

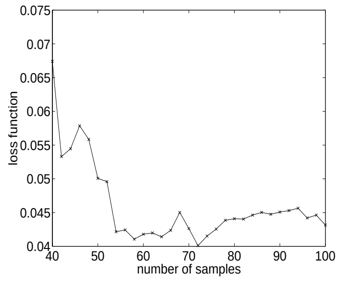


Fig. 7. Relation between loss function and the number of samples

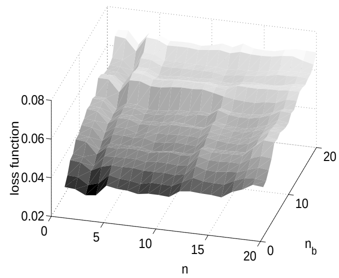


Fig. 8. Relation between loss function and the orders of the ARX model

choice of the orders and the delay of the ARX model, and the number of samples used for system identification. It is therefore desirable to carefully choose ζ and the number of samples to minimize the loss function $V_N(\theta, Z^N)$ (i.e., the sum of all squared prediction errors).

Figure 7 shows the relation between the loss function $J_N(\theta)$ and the number of samples used from the input and output data in the network configuration **N2**. In this figure, the orders and the delay of the ARX model is fixed at $\zeta = [8, 8, 1]$, while the number of samples is changed from 40 to 100. This figure shows a tendency that, as the number of samples increases, the loss function first decreases and then gradually increases. The similar tendencies are observed in other network configurations **N1** and **N3**, although the results are not included here.

We next show the relation between the orders of the ARX model and the loss function $V_N(\theta, Z^N)$ in Fig. 8. This figure uses 100 samples from the input and output data in the network configuration **N2**, and the orders of the ARX model, n_a and n_b, are changed from 1 to 20, respectively. This figure indicates that the loss function increases as the n_b increases. On the contrary, the choice of n_a has little effect on the loss function.

Another important factor in determining the orders of the ARX model is the highest frequency that should be captured by the ARX model. Namely, the ARX model is able to capture higher frequency of the output data (i.e., the round-trip time variation) with larger n_a and n_b. Moreover, the ARX model requires more computational burden and becomes less stable as the orders increase [7]. So the orders of the ARX model should be determined by taking account of a trade-off among accuracy, complexity, and stability.

6 Discussion and Conclusion

We discuss several possible applications of our approach — modeling the round-trip time dynamics of the Internet using the ARX model. Details of these topics will be discussed in the forthcoming paper, but it is worthwhile to discuss how our approach can be applied to various problems. The first and straightforward application would be to use our approach to *understand* the round-trip time dynamics of the Internet. We can analyze the round-trip time dynamics through the ARX model. Because the ARX model is one of LTI (Linear Time Invariant) models, various analysis techniques for LTI models in time- and frequency-domain can be utilized. The second application would be to *predict* the future round-trip time from the ARX model. As have shown in Section 5, the round-trip time of a packet is considerably disturbed by background traffic. Hence, it is difficult to predict the far future round-trip time. However, the ARX model can predict the near future round-trip time, which would be useful to, for example, QoS controlling mechanisms. As noted in Section 1, the third and possibly most important application would be to *design* a delay-based congestion control mechanism. Once the ARX model capturing the round-trip time is obtained, it would be possible to apply the optimal control theory to design an efficient delay-based congestion control mechanism. Congestion control of the Internet is a difficult problem because of its complexity such as heterogeneity of various network elements and non-negligible propagation delays. However, we believe that combination of the ARX model and the optimal control theory would help us to design a more efficient congestion control mechanism. We are currently working on designing a delay-based congestion control mechanism for stream video applications.

In this paper, we have proposed a novel approach to model the round-trip time dynamics of the Internet using system identification. The main idea is to model the network, seen by a specific source host, as a linear time-invariant ARX model. The input to the ARX model is the packet inter-departure time from the source host, and the output is the round-trip time variation measured at the source host. With the ICMP-based measurement method, we have collected three sets of the input and output data from real networks. Using the measurement results, we have determined coefficients of the ARX model, and have investigated how accurately the ARX model captures the round-trip time dynamics. We have found that the ARX model can capture the round-trip time dynamics when the network is moderately congested. We have also found that, when the network is not congested or the measured round-trip time is noisy, the ARX model fails to capture the round-trip time dynamics.

As a future work, it is important to validate effectiveness of our modeling approach for a through set of input and output data obtained from various network configurations. We are currentry measuring the input and output data in working LAN and WAN environments [23].

Acknowledgement

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. R. Jain, "A delay-based approach for congestion avoidance in interconnected heterogeneous computer networks," *ACM Computer Communication Review*, vol. 19, pp. 56–71, Oct. 1989.

2. L. S. Brakmo, S. W. O'Malley, and L. L. Peterson, "TCP Vegas: New techniques for congestion detection and avoidance," in *Proceedings of ACM SIGCOMM '94*, pp. 24–35, Oct. 1994.
3. J.-C. Bolot, "Characterizing end-to-end packet delay and loss in the Internet," *Journal of High-Speed Networks*, vol. 2, pp. 305–323, Dec. 1993.
4. S. B. Moon, J. Kurose, P. Skelly, and D. Towsley, "Correlation of packet delay and loss in the Internet," tech. rep., Department of Computer Science, University of Massachusetts, USA, Jan. 1998.
5. V. Paxson, "End-to-end Internet packet dynamics," in *Proceedings of ACM SIGCOMM*, pp. 139–152, Sept. 1997.
6. H. Ohsaki, M. Murata, and H. Miyahara, "Modeling end-to-end packet delay dynamics of the Internet using system identification," in *Proceedings of Seventeenth International Teletraffic Congress*, pp. 1027–1038, Dec. 2001.
7. L. Ljung, *System identification — theory for the user.* Englewood Cliffs, N.J.: Prentice Hall, 1987.
8. M. Allman and V. Paxon, "On estimating end-to-end network path properties," *ACM SIGCOMM '99*, pp. 263–274, Aug. 1999.
9. A. Adams, T. Bu, R. Caceres, N. Duffield, T. Friedman, J. Horowitz, F. L. Presti, S. B. Moon, V. Paxson, and D. Towsley, "The use of end-to-end multicast measurements for characterizing internal network behavior," *IEEE Communications*, pp. 152–159, May 2000.
10. V. Paxson, "End-to-end routing behavior in the Internet," in *ACM SIGCOMM '96*, pp. 25–38, May 1996.
11. R. Caceres, P. B. Danzig, S. Jamin, and D. J.Mitzel, "Characteristics of wide-area TCP/IP conversations," in *ACM SIGCOMM '91*, pp. 101–112, Sept. 1991.
12. H. Che and S.-Q. Li, "Fast algorithms for measurement-based traffic modeling," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 612–625, June 1998.
13. L. A. Kulkarni and S.-Q. Li, "Measurement-based traffic modeling: capturing important statistics," *Journal of Stochastic Model*, vol. 14, no. 5, 1998.
14. S.-Q. Li, S. Park, and D. Arifler, "SMAQ: A measurement-based tool for traffic modeling and queueing analysis, Part I - design methodologies and software architecture," *IEEE Communications Magazine*, vol. 36, pp. 56–65, Aug. 1998.
15. S.-Q. Li, S. Park, and D. Arifler, "SMAQ: a measurement-based tool for traffic modeling and queueing analysis, Part II - network applications," *IEEE Communications Magazine*, vol. 36, pp. 66–77, Aug. 1998.
16. Y. Zhang, V. Paxon, and S. Schenker, "The stationarity of Internet path properties: routing, loss, and throughput," tech. rep., ACIRI, May 2000.
17. O. Ait-Hellal, E. Altman, and T. Basar, "A robust identification algorithm for traffic models in telecommunications," in *Proceedings of IEEE CDC '99*, Dec. 1998.
18. A. Sang and S.-Q. Li, "A predictability analysis of network traffic," in *Proceedings of IEEE INFOCOM 2000*, pp. 342–351, Mar. 2000.
19. S. Hares, "Essential tools for the OSI Internet," *Request for Comments (RFC) 1574*, Feb. 1994.
20. F. Baker, "Requirements for IP version 4 routers," *Request for Comments (RFC) 1812*, June 1995.
21. J. Postel, "Internet control message protocol," *Request for Comments (RFC) 792*, Sept. 1981.
22. S. Savage, "Sting: A TCP-based network measurement tool," in *Proceedings of USENIX Symposium on Internet Technologies and Systems*, pp. 71–79, Oct. 1999.
23. H. Ohsaki, M. Morita, and M. Murata, "Measurement-based modeling of Internet round-trip time dynamics using system identification," submitted to *the Second IFIP-TC6 Networking Conference (NETWORKING 2002)*, Nov. 2001.