Active/Passive Combination-type Performance Measurement Method Using Change-of-measure Framework

Keisuke Ishibashi, Toshiyuki Kanazawa, and Masaki Aida
NTT Information Sharing Platform Laboratories, NTT Corporation
3-9-11 Midori-cho, Musashino-shi, Tokyo 180-8585, Japan

Abstract— We propose a performance measurement method that uses both active and passive measurement data to infer the actual performance seen by users. With this method, the performance for individual users, organizations or applications can also be estimated. An actual implementation of the proposed method is examined through simulation. We also evaluated the estimation accuracy with respect to both the measurement interval and the number of measurements.

I. Introduction

Performance measurement is crucial in controlling, managing, and provisioning networks. In general, methods of measuring network performance are divided into two types: active and passive .

Active methods monitor the performance of a network by sending probe packets and monitoring them. Many active monitoring tools have been developed to monitor network performance [1]. They generally monitor the performance of the probe packets sent periodically to infer the performance of users' packets.

If we can assume that active monitoring measures the time average of network performance and that the user traffic is Poissonian, then the performance experienced by the users and the actively measured performance will be the same. This well-known property is called PASTA (which stands for "Poisson Arrivals See Time Average"). It is known, however, that current Internet traffic exhibits burstiness and is not Poissonian, in general [2]. In this case, more user packets are transmitted during congested periods, which means that more user packets experience worse performance. Thus, the performance experienced by users may actually be worse than that measured by periodical active monitoring. On the other hand, Operation, Administration, and Mantainance (OAM) cells are standardized for fault and performance management in ATM networks. They are sent every fixed number of user cells and they measure the network performance. There are studies applying this mechanism to IP networks [3] [4]. With this mechanism, the performance statistics seen by probe packets agree with those seen by users. But this mechanism sends more probe packets as the user traffic volume grows, so more additional traffic will be injected during congestion periods. In addition, implementation of these mechanisms requires tight transmission control of probe packets.

Passive methods capture packets and determine the network performance using their data. For example, by comparing two sets of time-series data captured with monitoring devices deployed at ingress and egress of the network, we can determine the delay and loss of these packets. By measuring the network performance in a the passive way, we can measure the performance experienced by users. However, these methods require identification of each packet by its header or content, which is hard when the packet volume is huge, as in a high-speed network.

In this paper, we propose a new performance measurement method for estimating the actual network performance experienced by users. Our method combines both active and passive monitoring using easy-to-measure methods. It can estimate not only the mixed performance experienced by all users but also the actual performance for individual users, organizations, and applications. In addition, it is scalable and lightweight.

The rest of the paper is organized as follows. In section II, we give a mathematical formalization of the framework of our method. In section III, as an application of the method, we propose a simple method for estimating the actual delay experienced by users. We also show the validity of the method through simulation. In addition, we extend our method to estimate the performance experienced by an individual user. The accuracy of the estimation is investigated in section IV in terms of the measurement interval and the number of measurements. Finally, we conclude the paper in section V.

II. Proposed Measurement Method

Our measurement method, CoMPACT Monitor; change-of-measure based passive/active monitoring, is based on a change-of-measure framework. It is scalable and lightweight and enables accurate estimation of detailed characteristics of performance for individual users, organizations, and applications. The combination of simple measurements of both active and passive types enables the change-of-measure framework.

A. Estimation of User Performance

Let X be the measurement objective, *e.g.*, the delay for user packets, whose distribution function is P. The distribution of X is written as

$$\begin{aligned} \Pr(X > a) &= \int \mathbf{1}_{\{x>a\}} \, \mathrm{d}P(x). \\ &= \mathrm{E}_P\left[\mathbf{1}_{\{X>a\}}\right]. \end{aligned} \tag{1}$$

Let us consider how to estimate the distribution of X. Suppose there are n arrivals in a measurement period, *e.g.*, n packets arrived. $X(i)$ denotes the i-th value of X. Then an estimator $Z_X(n,a)$ of distribution (1) can be obtained by using $X(i)$ as follows:

$$Z_X(n,a) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{X(i)>a\}}. \tag{2}$$

Actually, if $X(i)$ is ergodic, then $\mathbf{1}_{\{X(i)>a\}}$ is also ergodic for arbitrary $a \in \mathbf{R}$. Thus,

$$\lim_{n\to\infty} Z_X(n,a) = \Pr(X > a) \text{ a.s.} \tag{3}$$

Suppose we have a situation in which it is difficult to measure $X(i)$ directly, and an estimate of its distribution cannot

0-7803-7632-3/02/$17.00 ©2002 IEEE

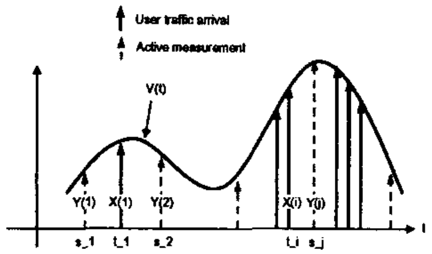

Fig. 1. Relationship between $V(t)$ and $X(i)$, $Y(j)$

be obtained with (2). Let $V(t)$ be the network performance at time t such that if the i-th arrival occurs at t_i, then $V(t_i) = X(i)$. Also, let Y be the value of $V(t)$ measured independently of the system behavior, and let the distribution function of Y be Q.

We consider estimating the distribution of X using the distribution of Y. We assume that for $a, b \in \mathbf{R}$,

$$P(b) - P(a) > 0 \Rightarrow Q(b) - Q(a) > 0. \tag{4}$$

Because P is the distribution of the network performance seen by packet arrivals and Q is the distribution of the measured network performance, this assumption indicates that the network performance that packets experience with a positive probability can be measured with a positive probability. This is natural when the measurement lasts long enough. and the independence of the measurement to the system behavior. We can then define $\mathrm{d}P/\mathrm{d}Q$ and rewrite the distribution of X given in (1) using the change-of-measure as

$$\begin{aligned} \Pr(X > a) &= \int \mathbf{1}_{\{y>a\}} \frac{\mathrm{d}P(y)}{\mathrm{d}Q(y)} \mathrm{d}Q(y) \\ &= \mathrm{E}_q \left[\mathbf{1}_{\{Y>a\}} \frac{\mathrm{d}P(Y)}{\mathrm{d}Q(Y)} \right]. \end{aligned} \tag{5}$$

Now, suppose Y is measured m times, and let $Y(j)$ be the j-th measurement at s_j such that $Y(j) = V(s_j)$ $(j = 1, 2, \ldots, m)$ (Fig. 1). Then an estimator of $\Pr(X > a)$ can be derived using $Y(j)$ as follows:

$$Z_Y(m, a) := \frac{1}{m} \sum_{j=1}^{m} \mathbf{1}_{\{Y(j)>a\}} L(j), \tag{6}$$

where

$$L(j) := \frac{\mathrm{d}P(Y(j))}{\mathrm{d}Q(Y(j))}, \tag{7}$$

which we call the likelihood ratio. Equation (3) also holds for $Z_Y(m, a)$ as

$$\lim_{m \to \infty} Z_Y(m, a) = \Pr(X > a) \text{ a.s.} \tag{8}$$

If we can derive $L(j)$, then the estimator of the distribution of X can be derived with the measurement values of Y. The fundamental concept of our method is as follows: Although estimation of the distribution $\Pr(X > a)$ from direct measurements of X is difficult, values $Y(j)$ and $L(j)$ can easily be measured by active and passive monitoring, respectively, and we can easily estimate the distribution $\Pr(X > a)$ using them. The derivation of likelihood ratio is described in the next subsection.

B. Likelihood Ratio

Let $\rho_X(t, \delta)$ be the traffic volume (*e.g.,* the number of packets) arriving in an interval $[t, t + \delta(t))$. Let $\rho_Y(t, \delta)$ be the number of measurements in $[t, t + \delta(t))$. We assume that the interval $\delta(t)$ is short enough compared with the time variance of $V(t)$ so that

$$V(s) \simeq V(s') \text{ for } \forall s, s' \in [t, t + \delta(t)). \tag{9}$$

This assumption indicates that one measurement of Y in the interval $[t, t + \delta(t))$ can be interpreted as $\rho_X(t, \delta)/\rho_Y(t, \delta)$ measurements of X. Note that we can always define $\rho_X(s_j, \delta)/\rho_Y(s_j, \delta)$ because at the time s_j of measurement $Y(j)$, we have $\rho_Y(s_j, \delta) > 0$. Then the likelihood ratio can be obtained as

$$L(j) = \frac{\rho_X(s_j) / \sum_{j=1}^{m} \rho_X(s_j)}{\rho_Y(s_j) / \sum_{j=1}^{m} \rho_Y(s_j)}. \tag{10}$$

The likelihood ratio (10) can be obtained by passive measurement, and the distribution of X is estimated as

$$Z_Y(m, a) = \frac{1}{\sum_{j=1}^{m} \rho_Y(s_j)} \sum_{j=1}^{m} \mathbf{1}_{\{Y(j)>a\}} \frac{\rho_X(s_j, \delta)}{\rho_Y(s_j, \delta)} \tag{11}$$

(Recall that $\sum_{j=1}^{m} \rho_X(s_j) = m$).

We can also derive an estimator of the mean of X, $M_Y(m)$, in a similar way to that for the distribution of X. This estimator is

$$M_Y(m) = \frac{1}{\sum_{j=1}^{m} \rho_Y(s_j)} \sum_{j=1}^{m} Y(j) \frac{\rho_X(s_j, \delta)}{\rho_Y(s_j, \delta)}. \tag{12}$$

C. Advantages of Our Method

We can expect our method to have the following advantages. (1) Since the extra traffic for active probe packets is negligible, user traffic is little affected. (2) We have a dependable estimation of the QoS/performance measure. (3) Since passive measurement is only required for counting the amount of traffic (the number of packets), the passive monitoring devices are simplified.

III. Delay Estimation

As an application of the method proposed in the previous section, we propose a simple method for estimating the actual delay experienced by users that is easy to implement.

A. Theoritical Basis

Let $Y(j)$ $(j = 1, 2, \ldots, m)$ be the delay measured with probe packets, such as ping, at time s_j. The probe packet interval $s_{j+1} - s_j$ is chosen to be a constant τ, and $\delta(s_j)$ is also chosen to be the same interval,for a simple implementation[1]. Then suppose that the number of user packets arriving in $[s_j, s_{j+1})$ is $\rho(j, \tau)$ and the total number of packets arriving in the measurement period is $\sum_{j=1}^{m} \rho(j, \tau) = n$. As an example of a delay estimation case, $V(t)$ is considered as the virtual waiting time of the network, which is the delay for a packet arriving (virtually) at t. If we assume that τ is short enough compared with the fluctuation of $V(t)$, then we derive the estimator of the packet-delay distribution by applying (11) for $\rho_X(s_j, \delta) = \rho(j, \tau)$, $\rho_Y(s_j, \delta) = 1$, and $\delta(s_j) = \tau$ as

[1] To eliminate influence of periodical network behavior, we can use exponentially distributed intervals.

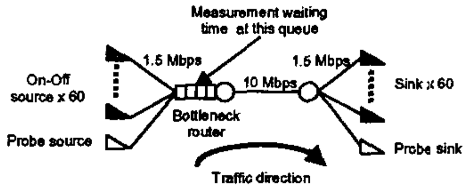

Fig. 2. Network configuration for simulation.

$$Z_Y(m,\tau,a) = \frac{1}{n}\sum_{j=1}^{m} \mathbf{1}_{\{Y(j)>a\}}\rho(j,\tau). \quad (13)$$

The estimator of the mean user packet delay, $M_Y(m,\tau)$, is also obtained as

$$M_Y(m,\tau) = \frac{1}{n}\sum_{j=1}^{m} Y(j)\rho(j,\tau). \quad (14)$$

As can be seen from (13), estimating the user delay requires measuring the network delay periodically with active probe packets and measuring the number of packets arriving between active measurements, which is far easier than measuring the delay for user packets directly with two probes deployed at the network edges.

The assumption that τ is short compared with the fluctuation of $V(t)$ is crucial for the estimation, so, we evaluate the relationship between the measurement interval τ and the estimation accuracy in section IV.

B. Evaluation

To demonstrate our simple application described above, we used the *ns2* [5] network simulator. Figure 2 shows the network topology for the simulation. Sixty sources were connected to a bottleneck router via 1.5-Mbps links, and two routers were connected via a 10-Mbps link[2].

We measured the queueing delay at the bottleneck router which did not include the service time for the packets themselves. Other simulation conditions are as follows:

- The user packets were generated by ON-OFF sources. We tested that the ON and OFF durations were distributed as i.i.d. exponentials and pareto. The mean ON duration was 1 s and the mean OFF duration was 14 s. For the pareto distribution, the shape parameter was chosen as 1.5.
- The user packet size was fixed at 1000 bytes.
- The transport protocol for the user packets was TCP.
- Probe packets for actively measuring the queueing delay were generated every second. The size of each probe packet was fixed at 64 bytes.

Figure 3 shows a sample path for the user packet delay, probe packet delay, and number of user packets arriving between probe packets for the case of exponential ON-OFF sources. It can be seen that the delay measured with probe packets captures the time variance of delay for the user packets well. However we can also see a fluctuation in the number of packets, which is synchronized with the delay fluctuation.

[2]In this paper, we only show the results for the single-bottleneck case. We have also evaluated our method for the multiple-bottleneck case, and found the bottleneck queueing delay to be the same as for the single bottleneck case, as can be expected from section II.

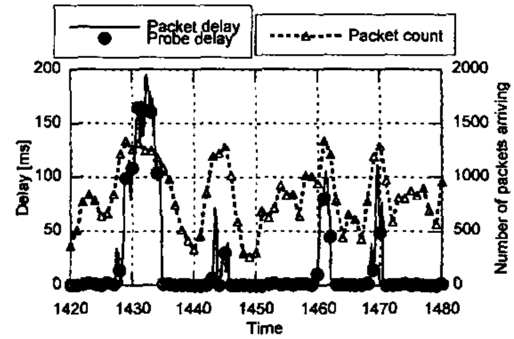

Fig. 3. Sample path for number of packets arriving and delay for user packets and probe packets in simulation.

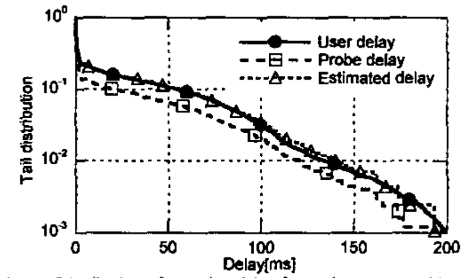

Fig. 4. Distribution of queueing delay for packets generated by exponential on-off sources, probe packets, and estimator.

This fluctuation causes the discrepancy between the distribution of delay for bursty user packets and periodical probe packets, because the number of packets with worse delay is larger for user packets than probe packets.

Figures 4 and 5 show the delay distributions of user packets, probe packets and an estimation. As expected from the sample path, we can observe the discrepancy between the distribution of user packet delay and that for active measurements. Using our proposed method, however, user delay can be estimated with high accuracy by active measurements.

C. Estimation of Individual User Delay

We describe here an extension of our measurement method, which estimates the packet delay for individual users with one series of active measurements and passive traffic monitoring.

Let X_k be the packet delay of user k ($k = 1, 2, \ldots, K$) and $Y(j)$ ($j = 1, 2, \ldots, m$) be the delay measured with active packets, such as ping, at time s_j. Let the number of packets for

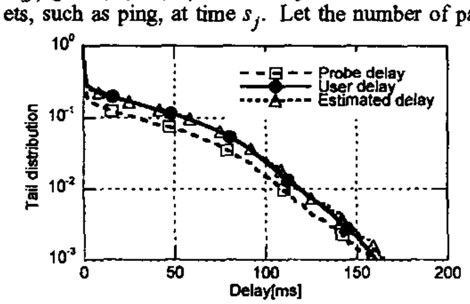

Fig. 5. Distribution of queueing delay for packets generated by Pareto on-off sources, probe packets, and estimator.

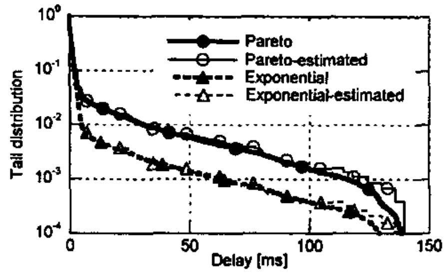

Fig. 6. Estimations of delay distribution for two type of user.

user k arriving in $[s_j, s_{j+1})$ be $\rho_k(j,\tau)$, and the number of total packets for user k is

$$n_k := \sum_{j=1}^{m} \rho_k(j,\tau). \tag{15}$$

Then the likelihood ratio for user k is

$$L_k(j) := \rho_k(j,\tau)\frac{m}{n_k}, \tag{16}$$

and we can obtain the estimator as follows:

$$Z_{Yk}(m,\tau,a) = \frac{1}{n_k}\sum_{j=1}^{m} \mathbf{1}_{\{Y(j)>a\}}\rho_k(j,\tau). \tag{17}$$

Thus, by counting the number of packets arriving for each user, we can estimate the delay experienced by individual users.

The classification of traffic is not limited to individual users or groups of users but can be extended to applications. The performance for packets may differ depending on the application because its traffic pattern differs depending on its application. Using our method, we can monitor the performance for each application with one series of active measurements, provided that packets for every class are treated with the same priority in the network.

We tested this extension by simulation. We separated the 60 sources from the simulation run in subsection III-B into 5 bursty sources and 55 non-bursty sources. For the bursty sources, the ON-OFF durations were distributed in a Pareto distribution with shape parameter of 1.5, where the mean ON duration was 1 second and the mean OFF duration is 14 seconds. For the non-bursty sources, the ON-OFF durations were distributed exponentially, where the mean ON duration was 10 seconds and the mean OFF duration was 5 seconds. The other parameters were the same as before.

Figure 6 shows the distribution of the user packet delay and the distribution estimated using (17). Our method could estimate the distributions of both groups of users with high accuracy from one series of active measurements.

IV. Estimation Accuracy

A. Relationship between estimation accuracy and probe intervals

Our estimation described in subsection III-A is based on the assumptions expressed in (9); that is, the delay of user packets is assumed to be almost the same throughout an interval τ. Hence, we expect the accuracy to improve with a shorter interval for active measurements. In this section, we analyze the accuracy of the method, especially for estimating the mean delay.

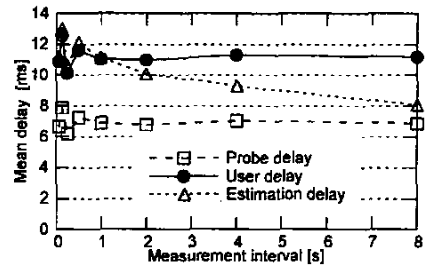

Fig. 7. Measured and estimated mean delay for various measurement intervals.

We ran simulations varying the measurement interval from 62.5 ms to 8 s and observed the changes in the discrepancy between the actual and estimated delay. For each measurement interval, five simulations were performed. The simulation conditions were the same as in subsection III-B except for the measurement intervals. We compared the mean delays for user packets, probe packets, and the estimator.

Figure 7 shows the results. The mean delay for user packets was about 11 ms, though that for probe packets was around 7 ms. In this case, the simple measurement using probe packets underestimate the queueing delay by about 40%. Also the estimator approximates the user delay with high accuracy when the interval is smaller than 1 s, which is the same as the mean ON duration. But as the interval increases, the estimation error increases.

Below, we consider the accuracy of the estimation in terms of the mean and the variance of the estimation error.

B. Mean of the Error

If the system can be assumed to be stationary, then we have

$$\begin{aligned} \mathrm{E}[M_Y(m,\tau)] &= \sum_{j=1}^{m}\mathrm{E}\left[Y(j)\frac{\rho_j}{n}\right] && (18)\\ &= m\mathrm{E}\left[Y(1)\frac{\rho_1}{n}\right] && (19)\\ &= \mathrm{E}[Y(1)] + \mathrm{Cov}[Y(1),\lambda_\tau(1)], && (20)\end{aligned}$$

where $\lambda_\tau(j)$ is defined as

$$\lambda_\tau(j) := \frac{m}{n}\rho_j, \tag{21}$$

whose mean is equal to 1. (Hereafter we simply write $\mathrm{E}[Y]$ or $\mathrm{Cov}[Y,\lambda_\tau]$ instead of $\mathrm{E}[Y(j)]$ or $\mathrm{Cov}[Y(j),\lambda_\tau(j)]$.)

We also have

$$\mathrm{E}[M_Y(n)] = \mathrm{E}[Y] + \lim_{\tau\to 0}\mathrm{Cov}[Y,\lambda_\tau]. \tag{22}$$

if we can assume that the probe packet is small enough. $\mathrm{E}[Y]$ is the mean delay measured with probe packets and is independent of the measurement interval. When $\lim_{\tau\to 0}\mathrm{Cov}[Y,\lambda_\tau]$ is zero, $\mathrm{E}[Y]$ agrees with the mean delay of user packets. The estimation error is evaluated from the difference between $\mathrm{Cov}[Y,\lambda_\tau]$ and $\lim_{\tau\to 0}\mathrm{Cov}[Y,\lambda_\tau]$.

Figure 8 plots the mean of $\mathrm{Cov}[Y,\lambda_\tau]$ and 95% confidence intervals for the five simulations. We can see that the covariance decays as the measurement interval increases. This is natural because the correlation between traffic intensity measured at a fixed interval and queueing delay decreases as measurement interval increases. We can also see that the covari-

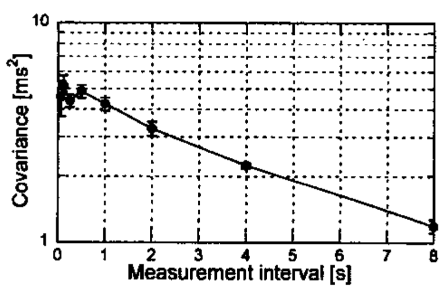

Fig. 8. Covariance between packet count and measured delay

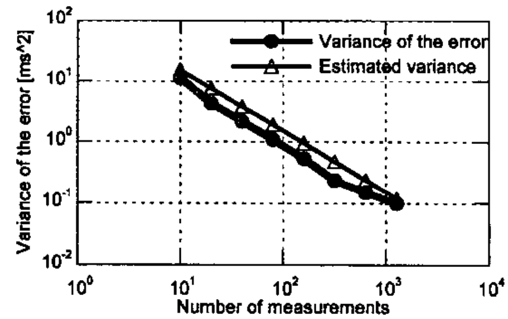

Fig. 9. Variance of the error against the various number of measurements

ance converges to a constant value, which is expected to be the $\lim_{\tau\to 0}\mathrm{Cov}[Y,\lambda_\tau]$, as the interval approaches zero. Thus, we can roughly estimate the appropriate measurement interval for required accuracy. For example, if we require 2-ms accuracy and assume $\lim_{\tau\to 0}\mathrm{Cov}[Y,\lambda_\tau]$ is about 5 ms, then measurements every 2 s are enough to achieve required accuracy. Note that, in this estimation, we use only available measurement values such as packet counts and probe delays.

C. Variance of the Error

The estimation error can be written as

$$M_Y(m,\tau) - M_X(n) = \frac{1}{n}\sum_{j=1}^{m}\left(\sum_{i=1}^{\rho(j)}\left(Y(j) - X^j(i)\right)\right), \quad (23)$$

where $X^j(i)$ is the delay of the i-th user packet arriving in the j-th measurement period.

First, we evaluate the variance of the value in the second bracket in (23). We define the conditional variance of the value for $\rho(j)$, $E_c(\rho(j))$, as

$$E_c(\rho(j)) \;:=\; \mathrm{Var}\left[\sum_{i=1}^{\rho(j)}\left(Y(j) - X^j(i)\right)\,\middle|\,\rho(j)\right]. \quad (24)$$

We also define the conditional variance and auto-covariance functions of $V(t)$ for $\rho(j)$ as $V_{\rho(j)}(t)$ and $C_{\rho(j)}(t)$, respectively. For a rough evaluation, we assume $\rho(j)$ user-packets arrive in the constant intervals between probe packets. We also assume the system to be stationary. Then

$$\begin{aligned} V_c(k) &:= \mathrm{Var}\left[\sum_{i=1}^{k}\left(V(0) - V\left(\frac{i\tau}{k+1}\right)\right)\right] \quad (25)\\ &\simeq (k^2+k)\,\mathrm{Var}\left[V_k(0)\right] \\ &- 2\sum_{i=1}^{k} i\,C_k\left(\frac{i\tau}{k+1}\right). \end{aligned}$$

Therefore, the unconditional variance of the sum of the errors for one measurement can be calculated with the distribution of the number of user packets $P_k = \Pr[\rho(j) = k]$ as:

$$\mathrm{Var}\left[\sum_{i=1}^{\rho(j)}\left(Y(j) - X^j(i)\right)\right] \simeq \sum_{k=1}^{\infty} V_c(k) P_k \quad (26)$$

If we can also assume that the sums of the error for different measurements are independent each other[3], then

[3]We confirm this independence from the auto-correlation function whose values are almost zero for both exponential and Pareto distributed On duration.

$$\mathrm{Var}\left[M_Y(m,\tau) - M_X(n)\right] \simeq \frac{m}{n^2}\sum_{k=1}^{\infty} V_c(k) P_k, \quad (27)$$

It may be observed from (27) that, if the measurement interval is fixed, then the variance of the error is inversely propotional to the number of measurements (Note that n increases linealy as m increases).

Note that, we can obtain the conditional auto-covariance function $C_{\rho(j)}(t)$ and the distribution P_k from the measurements of packet counts and probe delays. Figure 9 shows a plot of the variance of the estimation error and the estimated variance of the estimation error using (27) against the number of measurements. The simulation condition is the same as in III-B, and to obtain $C_{\rho(j)}(t)$ and $V_{\rho(j)}(t)$, we ran the simulation with probe interval as 100 ms, which is ten times shorter than the normal measurement. We can see that the variance decays as the number of measurements grows and the variance calculated from (27) estimates the variance of the error fairly well.

V. Conclusion

In this paper, we proposed a performance measurement method called *CoMPACT Monitor; change-of-measure based passive/active monitoring*, which can estimate user performance in a scalable and lightweight manner. We validated this method by simulations, which showed that our method gives a good estimation of the performance seen by a user. We extended this method to estimate individual user performance, and confirmed the validity of this approach by simulation. We also tested the applicability of the method in terms of estimation accuracy. We found that the mean of the estimation error depends on the measurement intervals and the variance of the error depends on the number of measurements. We have implemented our method and are now evaluating it on a real environment.

References

[1] CAIDA cooperative association for internet data analysis. `http://www.caida.org/tools/`.

[2] V. Paxon and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," IEEE/ACM Trans. on Networking, June 1995.

[3] T. Lindh, "A Framework for embedded monitoring of QoS parameters in IP-based virtual private networks," Proc of PAM2001, April 2001.

[4] M. Beige, R. Jennings, and D. Verma, "Low overhead continuous monitoring of IP network performance," Proc. of SPECTS'99, July 1999.

[5] UCB/LBNL/VINT Network Simulator - ns (version 2). `http://www.isi.edu/nsnam/ns`.

[6] R. W. Wolff, "Stochastic modeling and the theory of queues," Prentice Hall, 1988.