On-line Estimation by Importance Sampling for the Tail Probability of FIFO Queue Length

Nobuhiro Kobayasahi
eTRUST
Nagaoka, Niigata, Japan
Email: no-kobayashi@etrust.ne.jp

Kenji Nakagawa
Nagaoka University of Technology
Nagaoka, Niigata, Japan
Email: nakagawa@nagaokaut.ac.jp

***Abstract*—In this paper, we propose a new method of IS simulation by NS-2 for the tail probability of FIFO queue length. We do not increase the arrival rate of packet traffic, but decrease the service rate for queueing packets to increase the queue length. Then we propose an on-line estimation for the tail probability. We compare the simulation results by MC method, conventional IS method and our proposed IS method.**

I. INTRODUCTION

We consider the estimation using a network simulator NS-2 for the tail probability of an FIFO (First In First Out) queue length.

When performing the tail probability simulation using NS-2, the Monte Carlo (MC) simulation method is commonly used. Applying the MC method in NS-2, it is possible to directly utilize the stochastic structure of the simulation model and to create a program easily. However, in the case of the tail probability smaller than 10^{-5}, which is defined by ITU-T Rec.Y.1541[3] as a value for the most demanding classes, the tail events of interest do not occur frequently, the simulation takes much time, and the reliability of the estimate is low.

Then, we will apply the Importance Sampling (IS) simulation method in order to improve the accuracy of the estimate and a speed-up of simulation by NS-2. IS method is a kind of simulation methods that generates events of small probability more frequently using a probability distribution that is different from the original probability distribution. Then an unbiased estimate for the target probability is given by correcting the obtained values[1]. The authors[4] applied the IS method using NS-2 for the tail probability estimation of DRR (Deficit Round Robin) scheduler, and evaluated its performance.

Typically, in IS simulation, the packet arrival rate is increased to generate the tail events more frequently than MC method. However, is it possible to increase user's packet arrival rate in a practical simulation or a real network? The answer is "no" if we perform an on-line and real time estimation by using user's packet traffic as it is.

We would like to perform a fast simulation without modifying the use traffic. If it is possible, then we can estimate the tail probability in real time and use the obtained value for the admission control and the rate control of the packet transmission.

An important point of the IS method is to generate a large amount of tail events. It is possible to generate tail events by decreasing the processing rate rather than increasing the arrival rate of packets. It is a feature of decreasing processing rate that does not change the user traffic, so, an IS method can be realized by the processing of only network side. More specifically, increasing the queue length is performed by a slow packet processing virtually on the network side in parallel with the actual packet processing. We realize this mechanism as an additional counter to the NS-2. In order to perform a slow packet processing, we will not take a method providing another new timer. Taking advantage of NS-2 as an event driven simulator, the queue length is incremented or decremented only at the packet arrival (enqueue) event to FIFO queue or packet departure (dequeue) event from the queue.

The following three points are the objectives of the present study. (I) Estimating the tail probability of an FIFO queue by IS simulation using NS-2. (II) Generating more tail events than the MC method by decreasing the processing rate rather than increasing the arrival rate. Then, an on-line estimation is available. (III) Compare the simulation results of MC method and our IS method.

II. IS SIMULATION FOR QUEUES

Let us consider a common way to apply the IS method to queues. Denote by Q the queue length in the steady state and consider the IS method for the probability $P(Q > q)$, where q is some value. $P(Q > q)$ is called the tail probability of the queue length. The event $\{Q > q\}$ is called a tail event. If q is large, then the tail event $\{Q > q\}$ does not occur frequently, so we will use another distribution P' different from the original distribution P to generate the tail event $\{Q > q\}$ more frequently. The obtained value is modified to have an unbiased estimate for $P(Q > q)$ by the weighting function [1]. We call P' a simulation distribution because P' is used for simulation.

In the IS method for queues, the estimation for the tail probability $P(Q > q)$ is performed on a cycle basis[2]. A $cycle$ is the time interval during which the queue length is positive.

A. IS Estimate for Tail Probability of Queue Length

Denote by $t(1), t(2), \cdots, t(j), \cdots$ the departure epochs of packets from the queue where the queue length is governed by the original distribution P. Denote by $Q'(t(j))$ the queue length at the time $t(j)$ by the simulation distribution P'. Then, an IS estimate $\hat{P}_{\text{IS}}$ for $P(Q > q)$ is given as follows[2];

$$\hat{P}_{\text{IS}} = \frac{1}{D} \cdot \frac{1}{M} \sum_{m=1}^{M} \sum_{j=1} 1_q(Q'(t(j)))W(j), \tag{1}$$

$$D = \frac{1}{\tilde{M}} \sum_{m=1}^{\tilde{M}} \sum_{j=1} W(j), \tag{2}$$

$$W(j) = \frac{P\big(Q'(t(1)), \cdots, Q'(t(j))\big)}{P'\big(Q'(t(1)), \cdots, Q'(t(j))\big)}. \tag{3}$$

Here, M and $\tilde{M}$ represent the number of cycles. $1_q(Q'(t(j)))$ denotes the indicator function of the event $\{Q'(t(j)) > q\}$, that is 1 if $Q'(t(j)) > q$, and 0 otherwise. $W(j)$ is the weighting function and D is the IS estimate for the average cycle length. In (1), the part of $\hat{P}_{\text{IS}}$ excepting $1/D$ is the IS estimate for the time length during which the queue length exceeds q in a cycle [2]. In addition, in (1), the sum about j is considered as follows. In an IS method for queues, the queue length diverges as the time if the traffic intensity by P' is greater than 1. Then, in (1) for $j = 1, 2, \cdots$, if we observe enough number of j's for which $1_q(Q'(t(j))) = 1$, or $Q'(t(j)) > q$, then stop the observation and abort the cycle. Here, "enough number" means the number that we can recognize the statistic $\sum_{j=1} 1_q(Q'(t(j)))W(j)$ sufficiently converged. This number is about 20 empirically. This method of aborting a cycle in the middle is called the *dynamic IS* [2].

B. The Optimal Simulation Distribution

Denote by $Q(t(j)),\ j = 1, 2, \cdots$ the queue length by the original distribution P at the departure epoch $t(j)$. Suppose $Q(t(j))$ forms a Markov chain and it is represented as

$$Q(t(j)) = Q(t(j-1)) + X,\ j = 1, 2, \cdots, \tag{4}$$

with an i.i.d. random variable X. Then, the queue length is determined by X. The optimal X^* that achieves the minimum variance $V[\hat{P}_{\text{IS}}]$ of the estimate $\hat{P}_{\text{IS}}$ is given as follows[1].

Let us denote by $\varphi_X(\theta) \equiv E(e^{\theta X}) = \sum_x e^{\theta x} P(X = x)$ the moment generating function of X, and define the exponential change of measure for $P(X = x)$ by

$$P(X' = x) = \frac{e^{\theta x} P(X = x)}{\varphi_X(\theta)},\ \theta \in \mathbb{R}. \tag{5}$$

The following theorem is known[1].

[Theorem A] The random variable $X' = X^*$ that minimizes $V[\hat{P}_{\text{IS}}]$ is given by the exponential change of measure $P(X^* = x) = e^{\theta^* x} P(X = x)$, where θ^* is the solution of the equation $\varphi_X(\theta^*) = 1,\ \theta^* > 0$.

In what follows, the simulation distribution that minimizes the variance $V[\hat{P}_{\text{IS}}]$ is called the *optimal* simulation distribution.

III. Modeling of FIFO Queue

In this section, we describe models for the packet arrival to the FIFO queue and packet processing in the unit of byte.

A. Modeling of Packet Arrival

For packets arriving to the queue, we assume that there are K kinds of packet lengths l_k [Byte], $k = 1, \cdots, K$. Denote by Λ_k [Byte/sec] the arrival rate of packets with packet length l_k. We do not take the number of flows into consideration, but only consider the arrival rate of all the packets with packet length l_k in the traffic obtained by multiplexing. We assume that the packets with packet length l_k follow the Poisson process at arrival rate Λ_k / l_k [packet/sec].

B. Modeling of Packet Processing

Arrived packets are stored in a buffer of size ∞ and they are processed in the order of arrival. The service time for a packet is proportional to its packet length, i.e., the serive time for one byte in a packet is constant. The service rate is denoted by μ [Byte/sec]. Based on the above definitions, we consider that an FIFO queue is prescribed by the parameters

$$S = (\Lambda_k, \mu)_{k=1,\cdots,K}. \tag{6}$$

C. Probabilistic Structure of Variation of Queue Length

Denote by $Q_k(t)$ [Byte], $k = 1, \cdots, K$, the total number of bytes of all packets with packet length l_k in the queue at time t, and write $Q(t) = \sum_{k=1}^{K} Q_k(t)$ [Byte]. Denote by $t_d(1), t_d(2), \cdots, t_d(j), \cdots$ the departure epochs of packets from the queue. Here, the subscript d represents the $d$equeue from the queue. Denote by $t_d(0)$ the starting time of processing for the packet that departs at time $t_d(1)$.

Let $L(j)$ [Byte] be the packet length of the packet processed in the time interval $t_d(j-1) < t \leq t_d(j)$. The number of packets with packet length l_k arriving in the time interval $t_d(j-1) < t \leq t_d(j)$ is denoted by $n_k(j)$ and put $A(j) = \sum_{k=1}^{K} n_k(j) l_k$. $A(j)$ represents the total number of bytes that arrive in the interval $t_d(j-1) < t \leq t_d(j)$. Then, the following recurrence formula holds;

$$Q(t_d(j)) = Q(t_d(j-1)) + A(j) - L(j),\ j = 1, 2, \cdots. \tag{7}$$

Now, let us consider the difference of queue lengths $Q(t_d(j)) - Q(t_d(j-1)) = A(j) - L(j)$. By omitting the variable j, write $A = A(j)$, $L = L(j)$, and $X = A - L$. The maximum packet length is denoted by $l_{\max} = \max(l_1, \cdots, l_K)$. We will calculate the probability $P(X = x),\ x \geq -l_{\max}$, and the moment generating function $\varphi_X(\theta)$ of X.

Since we assumed that the buffer size is infinite, arrived packets are eventually processed by the server, not discarded. Therefore, the ratio of packets to be processed is proportional to the packet arrival rate Λ_k / l_k. Putting $\pi_k = P(L = l_k)$, we have

$$\pi_k = \frac{\Lambda_k / l_k}{\sum_{k'=1}^{K} \Lambda_{k'} / l_{k'}},\ k = 1, \cdots, K. \tag{8}$$

Denote by $\tau_k = l_k / \mu$ [sec] the processing time for a packet with length l_k. Then, the arrival rate $\lambda_{kk'}$ of packets with length $l_{k'}$ during a time interval τ_k is given as follows;

$$\lambda_{kk'} = \tau_k \frac{\Lambda_{k'}}{l_{k'}} = \frac{l_k}{\mu} \frac{\Lambda_{k'}}{l_{k'}}. \tag{9}$$

Further, define

$$f(\theta) = -\theta + \frac{1}{\mu}\sum_{k=1}^{K}\frac{\Lambda_k}{l_k}(e^{l_k\theta} - 1). \tag{10}$$

For the sake of simplicity of symbols, we write the Poisson probability as $R(n;\lambda) = \lambda^n e^{-\lambda}/n!$ and $n \cdot l = \sum_{k=1}^{K} n_k l_k$.

Using above defined symbols, we have [4]

$$P(X = x) = \sum_{k=1}^{K}\pi_k \sum_{n\cdot l = x + l_k}\prod_{k'=1}^{K} R(n_{k'};\lambda_{kk'}), \tag{11}$$
$$x \geq -l_{\max},$$

$$\varphi_X(\theta) = \sum_{k=1}^{K}\pi_k \exp\big(f(\theta)l_k\big). \tag{12}$$

Then, we see that there uniquely exists $\theta = \theta^*$ with

$$\varphi_X(\theta^*) = 1,\ \theta^* > 0, \tag{13}$$

see [4].

D. The Optimal Simulation Distribution for FIFO Queue

We will determine the optimal simulation distribution that minimizes the variance of the IS estimate $\hat{P}_{\text{IS}}$ for the tail probability of an FIFO queue. The solution is given by [4] as

$$P(X^* = x) = \sum_{k=1}^{K}\pi_k \sum_{n\cdot l = x + l_k}\prod_{k'=1}^{K} R(n_{k'};\lambda^*_{kk'}), \tag{14}$$
$$x \geq -l_{\max},$$
$$\lambda^*_{kk'} = e^{l_{k'}\theta^*}\lambda_{kk'},\ k, k' = 1, \cdots, K, \tag{15}$$

where θ^* is given by (13).

IV. On-line IS Estimation

Now, let us consider (15). Denote by $S = (\Lambda_k, \mu)_{k=1,\cdots,K}$ the system parameters on the queue length variation rule X in (4) of the original FIFO queue. Then we write the system parameters on the optimal queue length variation rule X^* in the IS method as follows;

$$S^* = (\Lambda^*_k, \mu^*_k)_{k=1,\cdots,K}\,. \tag{16}$$

Similar to (9), we have

$$\lambda^*_{kk'} = \frac{l_k}{\mu^*_{k'}}\frac{\Lambda^*_{k'}}{l_{k'}}, \tag{17}$$

thus by (9), (15), (17), we have

$$\frac{\Lambda^*_k}{\mu^*_k} = e^{l_k\theta^*}\frac{\Lambda_k}{\mu},\ k = 1, \cdots, K. \tag{18}$$

In the IS method, if the arrival rate Λ^*_k [Byte/sec] and the processing rate μ^*_k [Byte/sec] satisfies the relation (18), then S^* in (16) is the optimal parameter. There is some degree of freedom for the determination of the parameters Λ^*_k, μ^*_k by (18). In particular, in [4], the authors put $\Lambda^*_k = e^{l_k\theta^*}\Lambda_k$, $\mu^*_k = \mu$. Their method is that the queue length is increased by increasing the arrival rate, which is the same as conventional IS methods. In this paper, IS method is performed by

$$\Lambda^*_k = \Lambda_k,\ \mu^*_k = e^{-l_k\theta^*}\mu,\ k = 1, \cdots, K. \tag{19}$$

The characteristic of the parameter setting by (19) is that $\Lambda^*_k = \Lambda_k$, i.e., the user traffic is not changed. By setting $\mu^*_k = e^{-l_k\theta^*}\mu$, the queue length is increased by decreasing the processing rate.

Now, we call the queue length in the IS method the *IS queue length*. In particular, the IS queue length by the optimal parameter (19) is called the optimal IS queue length, and denoted by Q^*. We will propose the following counters to calculate the Q^*.

A. On-line Calculation of IS Queue Length

As mentioned in the previous section, the characteristics of the IS simulation in this paper is that a large number of tail events are generated by decreasing the processing rate without changing the arrival rate. Because we do not change the user traffic, the IS method can be realized only by the modification of processing on the network side. Therefore, unlike the conventional IS methods, our IS can be performed on-line.

In order to decrease the processing rate, one may consider the use of a new timer to manage the processing time for slow packet processing. However, the system becomes complex if such a new timer is implemented in NS-2. In this paper, we propose a counter that can count the optimal IS queue length Q^* easily, by utilizing the feature of NS-2 operating in event driven manner.

B. ISQL Counter and its Algorithm

We will prepare the following counters for calculating the optimal IS queue length.

Q^* [Byte]	counter for the optimal IS queue length
$\{C_k\}_{k=1\cdots,K}$	counter for slow packet processing

The set of these counters $\{Q^*, \{C_k\}_{k=1,\cdots,K}\}$ is called the ISQL counter (IS Queue Length counter).

We will explain the ISQL counter. Denote by $t_e(1)$, $t_e(2)$, $\cdots$, $t_e(i)$, $\cdots$ the arrival epochs of packets to the FIFO queue, and $t_d(1), t_d(2), \cdots, t_d(j), \cdots$ the departure epochs from the queue. The indices e and d stand for *e*nqueue and *d*equeue, respectively. In addition, let PktInd(t) be a function that returns the index of the packet arriving or departing at time t. Here, the index of the packet denotes the subscript k of the packet with packet length l_k.

We will show in Figure 1 a pseudo code of the algorithm of ISQL counter.

The algorithm of ISQL counter

1 at time 0
2 $Q^* = 0$:
3 $C_k = 0, \ \ k = 1, \cdots, K$
4 at enqueue epoch $t_e(i)$
5 let $k =$PktInd$(t_e(i))$:
6 $Q^* = Q^* + l_k$
7 at dequeue epoch $t_d(j)$
8 let $k =$PktInd$(t_d(j))$:
9 $C_k = C_k + 1$
10 If $C_k \geq \exp(l_k\theta^*)$
11 then $Q^* = Q^* - l_k$:
12 $C_k = C_k - \exp(l_k\theta^*)$

Fig. 1. Pseudo Code of the Algorithm of ISQL Counter

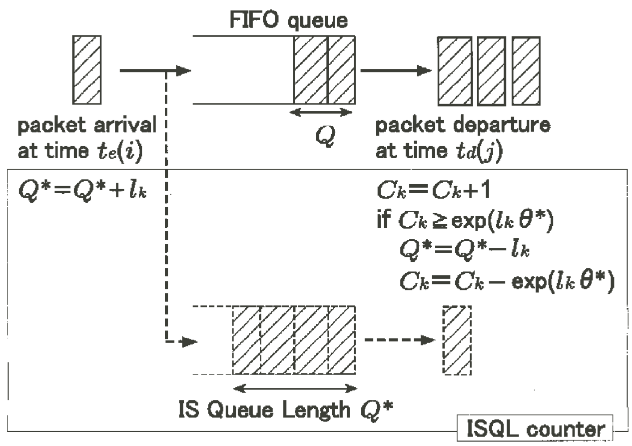

Fig. 2. Decrease of service rate and IS queue length by ISQL counter

Let us explain the operation of the above algorithm. First, as a basic thing, this operation is performed in cycles on the IS queue length Q^*. Here, a cycle is defined as a time interval during which $Q^* > 0$.

In lines 1-3 in Figure 1, the IS queue length Q^* and the counters C_k are initialized at time 0. In lines 4-6, at the arrival epoch $t_e(i)$ to the FIFO queue (enqueue epoch), the packet length l_k is added to the IS queue length Q^*. This operation at the arrival epoch is the same as the operation on the queue length of the FIFO queue. In line 9, at the departure epoch (dequeue epoch) $t_d(j)$ from the FIFO queue, 1 is added to the counter C_k corresponding to the departing packet. In line 10, if the counter value C_k becomes greater than or equal to $\exp(l_k\theta^*)$, in line 11, l_k is subtracted from the IS queue length Q^*, and in line 12, $\exp(l_k\theta^*)$ is subtracted from C_k. The operations in lines 9-12 are that the counter C_k is incremented one by one and if C_k reaches $\exp(l_k\theta^*)$ then the packet size l_k is subtracted from Q^*, so the processing time for a packet with length l_k becomes about $\exp(l_k\theta^*)$ times longer, in other words, the processing rate becomes about $\exp(-l_k\theta^*)$ times smaller. Because of this reason, the IS queue length Q^* quickly increases. In the case that $\exp(l_k\theta^*)$ is not an integer, a fraction occurs in the subtraction in line 12, but the fraction is carried forward to the next time, thus, by considering in long time, a packet with length l_k is processed at the rate about $\mu^* = \mu \exp(-l_k\theta^*)$.

It should be noted here that the transmission of packets with length l_k from the IS queue length Q^* is not necessarily performed in the order of arrival. The length of a packet is ignored in Q^* and only the total number of bytes of all packets are managed by the Q^*. This fact is indicated by a dotted line between packets in Q^* in Figure 2. Because our aim is to obtain the probability of the event $\{Q > q\}$, the order of packet processing does not affect the results.

C. IS Estimate by ISQL Counter

Using the optimal IS queue length Q^* obtained by the ISQL counter as the queue length Q' in (1),(2),(3), we have the following IS estimate $\hat{P}_{\text{IS}}$ for the tail probability $P(Q > q)$ in a similar way as [4];

$$\hat{P}_{\text{IS}} = \frac{1}{D} \cdot \frac{1}{M} \sum_{m=1}^{M} \sum_{j=1} 1_q(Q^*(t_d(j)))W(j), \tag{20}$$

$$D = \frac{1}{\tilde{M}} \sum_{m=1}^{\tilde{M}} \sum_{j=1} W(j), \tag{21}$$

$$W(j) = \exp\left(-\theta^* Q^*(t_d(j))\right). \tag{22}$$

The summation on j in (20),(21) is determined by the dynamic IS method, which aborts a cycle in the middle of the simulation, as mentioned in the last part of the section II-A.

V. Evaluation by Simulation

We will investigate the accuracy of the estimates for the tail probability of the FIFO queue, the limit of the estimates, and the simulation time. The simulation methods that we will compare are the MC method, the conventional IS method which increases the arrival rate, and our proposed IS method which decreases the processing rate. The purpose of the comparison of performance is to make sure that the basic performance of the proposed IS is equal to the conventional IS method, but not claim that our method is superior in these performance to the conventional IS method. Features of our on-line IS method proposed in this paper is the high-speed estimation for the tail probability by utilizing the user traffic as it is in the actually operating network, but it is difficult to claim the feature of “on-line” numerically. In this section, therefore, we will evaluate the basic performance of the proposed IS method as one of the IS methods.

A. Simulation 1

Simulation 1 is performed under the conditions shown in Tables I, II. In this simulation, we consider the case of $K = 2$, that is, there are two kinds of different packet lengths. The parameters of ISQL counter is shown in Table III. In Table III, for example in case 1, the optimal processing rates are $\mu_1^* = 32.67$[kByte/sec], $\mu_2^* = 8.54$[kByte/sec], i.e., the optimal processing rates are different for packets with short and long packet lengths.

TABLE I
PARAMETERS ON PACKET LENGTH

Number of different packet lengths	$K = 2$
Packet length[Byte]	$l_1 = 128$, $l_2 = 256$

TABLE II
PARAMETERS ON FIFO QUEUE AND CONVENTIONAL IS

	case 1	case 2	case 3	case 4
arrival rate Λ_1 [kByte/sec]	0.714	1.43	2.14	2.86
arrival rate Λ_2 [kByte/sec]	24.3	48.6	72.9	97.1
processing rate μ [kByte/sec]	125	125	125	125
traffic intensity $(\Lambda_1 + \Lambda_2)/\mu$	0.2	0.4	0.6	0.8
optimal arrival rate Λ_1^* [kByte/sec]	2.73	3.25	3.45	3.55
optima arrival rate Λ_2^* [kByte/sec]	355.8	249.3	190.3	150.5

The estimates of the tail probability are shown in Figure 3. We can see from this Figure that the proposed IS has equal performance to the conventional IS.

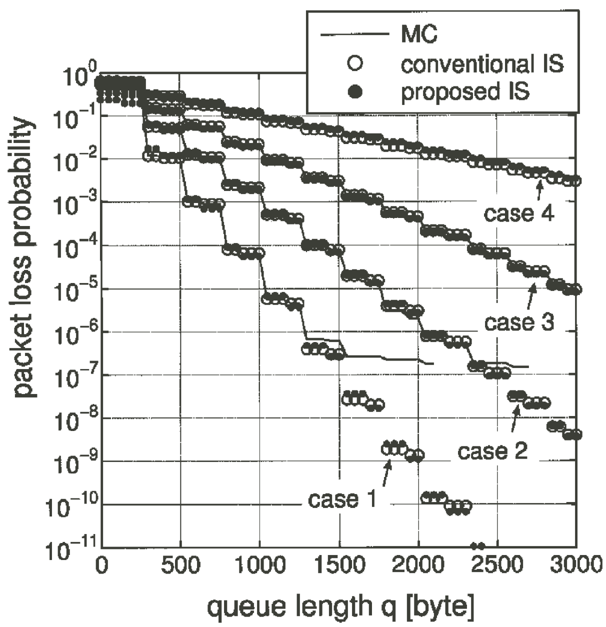

Fig. 3. Comparison of estimates by MC, conventional IS and proposed IS

B. *Simulation 2 [Evaluation of Simulation Time]*

We will compare the simulation time of the MC and the proposed IS in case 2 in the simulation 1 above, i.e., the time taken to estimate the tail probability $P(Q > q)$ for $q = 2000$ [Byte]. The number of cycles in one MC simulation is 10^8, while in the proposed IS it is 10^3. Under these numbers of cycles, the variances of the estimates are almost the same for MC and proposed IS. We see from Table IV that the simulation time is 379.8 [sec] by MC and 0.093 [sec] by the proposed IS. Thus, we achieved about 4,000 times speed-up.

TABLE III
PARAMETERS ON ISQL COUNTER

	case 1	case 2	case 3	case 4
θ^*	1.04 $\times 10^{-2}$	6.39 $\times 10^{-3}$	3.74 $\times 10^{-3}$	1.71 $\times 10^{-3}$
optimal processing rate $\mu_1^* = \mu e^{-l_1 \theta^*}$ [kByte/sec]	32.67	55.18	77.42	100.44
optimal processing rate $\mu_2^* = \mu e^{-l_2 \theta^*}$ [kByte/sec]	8.54	24.35	47.95	80.71
ratio of processing rates μ/μ_1^*	3.83	2.27	1.61	1.24
ratio of processing rates μ/μ_2^*	14.64	5.13	2.61	1.55
traffic intensity	2.53	1.92	1.51	1.22

TABLE IV
COMPARISON OF SIMULATION TIME BY MC AND PROPOSED IS

	MC	proposed IS
number of cycles	10^8	10^3
average of estimates	3.03×10^{-6}	2.67×10^{-6}
sample variance	1.47×10^{-14}	5.01×10^{-15}
simulation time	379.8 [sec]	0.093 [sec]

C. *Conclusion*

In this paper, we have proposed an on-line IS simulation method to estimate quickly the tail probability of an FIFO queue.

In a conventional IS, in order to generate a large number of the tail events the arrival rate is increased. In this study, we have proposed a new IS method to generate a large number of tail events by decreasing the processing rate and accept the packet traffic from the user as it is. By this proposed method, an on-line estimation is carried out and the simulation time becomes shorter than the MC method. We installed our proposed IS simulation program in NS-2. For the on-line estimation, "ISQL counter" was placed in the FIFO queue and the tail events were generated by decreasing the packet processing rate taking the advantage of NS-2 as an event driven simulator. By performing various simulations, we evaluated the accuracy of the estimates, the limit of estimates, and the simulation time, and so on.

REFERENCES

[1] J.A.Bucklew, *Large Deviation Techniques in Decision, Simulation, and Estimation*, Wiley, New York.
[2] M. Devetsikiotis and J.K. Townsend, "Statistical Optimization of Dynamic Importance Sampling Parameters for Efficient Simulation of Communication Networks," IEEE/ACM Trans. Networking, vol.1, no.3, pp.293-305, June, 1993.
[3] ITU-T Recommendation Y.1541, http://www.itu.int/rec/T-REC-Y.1541-200602-I/en
[4] K. Izumi, K. Nakagawa and T. Yokotani, "Application of Importance Sampling Simulation to NS-2 for Packet Loss Probability of DRR Scheduler," IEICE Trans. on Commun. Vol.J94-B, No.5, May, 2011 (in Japanese).
[5] K. Ogawa and K. Nakagawa, "The Optimal Simulation Distribution of MMPP/D/1 Queueing," IEICE Trans. on Commun. Vol.J80-B-I, No.2, pp.64-73, 1997 (in Japanese).
[6] J.S. Sadowsky, "Large Deviations Theory and Efficient Simulation of Excessive Backlogs in a GI/GI/m Queue," IEEE Trans. Auto. Cont., vol.36, no.12, pp.1383-1394, 1991.