Opportunistic Link Overbooking for Resource Efficiency Under Per-Flow Service Guarantee

Jianming Liu, Xiaohong Jiang, *Senior Member, IEEE*, and Susumu Horiguchi, *Senior Member, IEEE*

***Abstract*—The trade-off between resource efficiency and Quality of Service (QoS) is always a vital issue for communication networks, and link overbooking is a common technique used to improve resource efficiency. How to properly overbook a link and analytically determine its overbooking factor under QoS constraints are still problems, especially when achieving advanced QoS by per-flow queueing, as urged by the emerging mobilized applications in the access networks. This paper first proposes an Opportunistic Link Overbooking (OLO) scheme for an edge gateway to improve its link efficiency, and then develops an integrated analytical framework for determining the suitable link overbooking factor with service guarantee on flow level. In our scheme, once the idle time of a high priority flow's quasi-dedicated link is larger than a specified threshold, the link is temporarily overbooked to a low priority flow; and then when the high priority flow's subsequent packets start arriving, the link can be recovered at the expense of a setup delay. To explore the balance between link efficiency and the flow's QoS in the proposed scheme, we develop the corresponding queueing model under either bounded packet delay (relevant to delay-sensitive flow) or finite buffer size (relevant to loss-sensitive flow). Our queueing analysis reveals the inherent trade-offs among the link overbooking factor, packet loss rate and delay/jitter under different traffic patterns.**

***Index Terms*—Resource efficiency, opportunistic link overbooking, per-flow service guarantee, delay-sensitive flow, loss-sensitive flow.**

I. Introduction

Resource Efficiency and QoS are the vital issues in any communication network. There is always a trade-off between the efficiency in the use of network resources and the strictness of service guarantee. The widely deployed service model, Differentiated Services (Diffserv) [1], uses per-class queueing/management to reduce the complexity of QoS mechanism, while adopting link overbooking in the edge gateways to avoid resource wasting.

There is hardly any clear definition on overbooking [2] [3], although the network operators have been using it for a long time. Usually, overbooking is interpreted and implemented in the networks differently. A general understanding of overbooking is to have the sum of the allocated bandwidth of flows on a link exceed the link's physical bandwidth so as to achieve multiplexing gain. Overbooking also implies that the resources reserved to one flow can possibly carry traffic for another flow.

Proper overbooking may reduce resource wasting, but aggressive overbooking leads to QoS degradation. The concept of Effective Bandwidth (EB) [4] [5] can be used to perform link overbooking under Diffserv service model. However, the EB of a flow is link dependent, varying with link bit rate and buffer size, so it is still hard to precisely determine the appropriate degree of link overbooking under QoS constraints.

In the Diffserv architecture, QoS is supported at a class level rather than flow level, where the flows within a class share the link with same priority. It is notable that to overbook a link on per-class basis, the congestion will hit all the flows in a class, even those with high priority. Hence, services to the traffic flows are expected to be guaranteed on flow level, as is also urged by the emerging versatile applications with diverse traffic profiles and service requirements (e.g., some flows are very delay-sensitive, others are very loss-sensitive).

Recent research [6] [24] show that per-flow queueing/management has attractive capability to guarantee the QoS on flow level, where a flow can reserve its private resources (referred as its dedicated link). The *link* represents the interconnecting channel or (virtual) circuit, associated with some resources (buffers, bit rate, codecs capacity, etc.), between two locations for the purpose of transmitting or receiving data. In the edge (access) networks, the links are managed by the gateways [7] [8], which are a kind of special router with some additional functions, such as the translations between different protocols, data formats and audio/video codecs. Setting up a link requires resource reservation and allocation, and also needs a certain amount of time.

The trade-off between a link's efficiency and its QoS capability is more severe when supporting per-flow QoS. The challenges lie in two folds. First, compared with the soft reservation of resources in Diffserv architecture, providing dedicated link to a flow is costly, although by this way, a high priority flow's service can be guaranteed. Recent progress on Dynamic Queue Sharing [6] makes per-flow link management feasible while maintaining robustness and scalability. Second, dedicated link reservation often results in low efficiency. The situation becomes more critical with the deployment of numerous emerging applications, e.g., network games and Web browsing/shopping, etc., which only produce traffic from time to time. Furthermore, in the increasingly mobilized networks, due to the limitation of battery technology, wireless

Paper approved by T. T. Lee, the Editor for Switching Architecture Performance of the IEEE Communications Society. Manuscript received August 22, 2008; revised June 24, 2009; October 23, 2009; and December 17, 2009.

This work was supported in part by the JSPS Grant-in-Aid of Scientific Research (B) 21300018, the National Natural Science Foundation of China (60762002), GuangXi Province (0731024), GuiJiaoKeYan 2006 (No.26, D200644), and the State Key Laboratory for Novel Software Technology, Nanjing University, China.

J. Liu is with the School of Computer and Control, GuiLin University of Electronic Technology, GuangXi, 541001, P.R. China (e-mail: jmliu@guet.edu.cn).

X. Jiang and S. Horiguchi are with the Department of Computer Science, Graduate School of Information Sciences, Tohoku University, Sendai 980-8579, Japan (e-mail: {jiang, susumu}@ecei.tohoku.ac.jp).

Digital Object Identifier 10.1109/TCOMM.2010.06.080435

nodes have to prolong their lifetimes by adopting energy-saving techniques, such as the sleep scheme [9], energy-aware traffic shaping/media transcoding [10] [11], and real-time packet compression [12]. Those techniques let a traffic source spend much of its time in *silent* state [6] [13] (i.e., no packets generated), which causes large inter-arrival times occur frequently in the flow, and thus introduces large and frequent idle periods into a dedicated link.

On the other hand, the legacy problem is that low priority flows often suffer bandwidth starvation when high priority traffic is intense [30] [31]. Therefore, overbooking mechanism must be carefully designed, especially the proper degree of link overbooking, when supporting per-flow QoS.

Various engineering approaches [2] [32] have been proposed for link overbooking in Diffserv architecture, where the EB and network calculus [32] are applied to determine the overbooking factor, i.e., to calculate the required bandwidth of flows and multiplex a certain amount of flows within a same class into one link. The EB, which is between the average and peak rate of the flows, describes the minimum bandwidth required to fulfill an expected service for a given amount of traffic. The network calculus is a worst-case analysis method, which can provide the upper bound of QoS.

Those overbooking approaches are customized for the Diffserv service model, and the fundamental concepts of EB and network calculus are not suitable for resource overbooking when adopting dedicated link reservation. Two problems still remain: 1) how to improve the efficiency of the dedicated link; 2) how to maintain the link's QoS capability; and few work has addressed the trade-off issues between them when developing per-flow QoS. Tremendous endeavors have focused on providing flow level service guarantee based on the Diffserv architecture (see, e.g., [14-19]), and much other work has put their emphasis on improving the efficiency and scalability by dynamic resource management (see, e.g., [6, 20-22]). Besides, seldom work has aimed at developing an integrated analytical framework on flow level, which is critical for seeking the balance between link efficiency and QoS guarantee to determine the proper link overbooking factor, while considering the different requirements of delay-sensitive and loss-sensitive flows.

This paper proposes an Opportunistic Link Overbooking (OLO) scheme for edge gateways to improve the efficiency of a flow's dedicated link, while guaranteeing the flow's QoS at the same time. In our scheme, once the idle time of a high priority flow's dedicated link is larger than a specified threshold (called *inactivity time*), the link is temporarily overbooked to a low priority flow; and then when the high priority flow's subsequent packets start arriving, the link can be recovered at the expense of a setup delay. Because here we use the dedicated link in a slightly different way, we refer it as a *quasi-dedicated link*. Link setup delay is involved because of recovering the buffers, circuit or the channel (e.g., restoring the VPI/VCI values in the ATM/MPLS networks), which will degrade the service of high priority flow. The inactivity time directly determines link overbooking factor, can be used to control the trade-off between link efficiency and service quality degradation. To facilitate the setting of link overbooking factor, we develop a queueing model for our scheme, whose server is with delayed vacation and setup time.

Notice that the edge gateways carry both delay-sensitive and loss-sensitive flows [23]. The former refers to real-time applications, such as voice over IP or network games, which can tolerate moderate packet loss but demand delay/jitter guarantee from the network, where the packets violating the delay constraint will be dropped. The latter includes Web browsing/shopping, business transaction data etc., which require zero packet loss but are insensitive to delay, where packet loss is mainly due to buffer overflow. Different service requirements need different resource dimensioning methods. To satisfy the stern delay/jitter requirements, the average statistics (such as mean delay) are no longer suitable performance measures. We thus bound the waiting time of the admitted packets when handling the delay-sensitive flow. While for supporting the loss-sensitive flow, the finite buffer condition has been working well. Hence, we apply two packet admission policies: 1) bounded packet delay; 2) finite buffer size, separately on our scheme and conduct their corresponding queueing analysis, coming up with the packet loss rate, delay/jitter of the delay-sensitive flow and the packet loss and mean delay of the loss-sensitive flow.

Main contributions of our work are summarized as follows:

1) We propose an opportunistic link overbooking scheme to improve the resource efficiency of the edge gateway supporting QoS on flow level.
2) We develop a vacation queueing model for the scheme and provide comprehensive queueing analysis, which can quantitatively evaluate the impact of overbooking on the flow's service quality, and thus find the proper link overbooking factor.
3) We provide two different resource dimensioning methods for delay-sensitive and loss-sensitive flows, and conduct the corresponding capacity analysis of a flow's dedicated link based on our queueing analysis.
4) Finally, we demonstrate the inherent trade-offs between overbooking factor and flow's QoS measures. The comparative studies between Poisson traffic and the more realistic traffic model (e.g., Markov-modulated rate process) show the effectiveness of our scheme. Our work also implies that, by dimensioning the transmission resources under bounded packet delay, the delay performance of a delay-sensitive flow can be guaranteed at the expense of moderate packet loss.

The results of our work can be used to achieve a graceful efficiency-QoS tradeoff and thus an efficient resource management for edge gateways. It must be noted that, the OLO scheme is more effective on bursty or highly-correlated traffic. To shed lights on the nature of OLO scheme, we have applied the Poisson traffic model in our analysis, which can provide enough insights on the scheme performance while also presenting a feasible engineering approach to determining the proper overbooking factor with moderate computation complexity.

After introducing the link overbooking scheme and its queueing model in Section II, we will analyze the performance of our scheme under bounded delay and finite buffer conditions in Section III and IV, respectively. The trade-off relationships among link overbooking factor and QoS

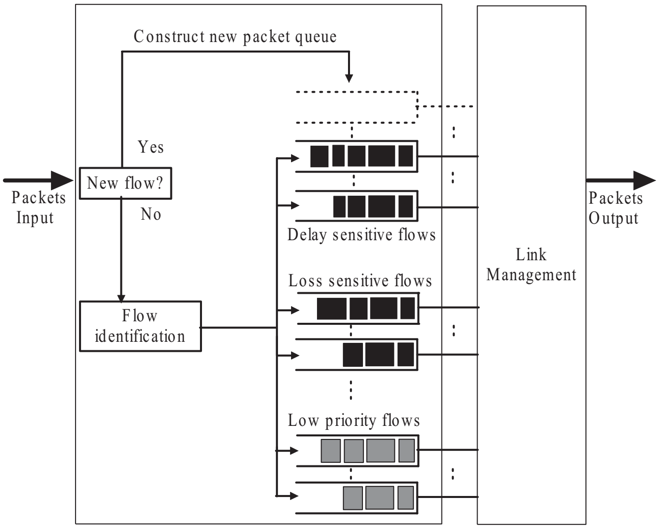

Fig. 1. Edge gateway adopting per-flow queueing.

measures are exhibited in Section V, and finally Section VI concludes the paper.

II. Opportunistic Link Overbooking Scheme and its Queueing Model

A. The OLO Overbooking Scheme

From the perspective of resource management, packet queues and links are essential components of gateways, where the incoming packets are buffered in the queue and cleared by the link. As depicted in Fig. 1, the edge gateway we consider adopts per-flow queueing policy [6], [24]. In the gateway, the input packet streams are classified into different packet flows by flow identification and an isolated queue is maintained for each flow. The flows can be divided into low priority flows and high priority flows. The former, such as FTP or Email flows, usually receives best effort service, having no stringent QoS requirement, but often suffering bandwidth starvation if high priority flows are heavy. The latter can be further classified as delay-sensitive flows and loss-sensitive flows, which demand service guarantee from the network and are the focus of this paper.

The link management module is responsible for managing the link of every packet flow according to the overbooking scheme depicted in Fig. 2. Without loss of generality, in our discussion we just focus on one of the packet queues (referred as tagged queue hereafter). For a better understanding of our scheme, the tagged packet queue can be regarded as having a virtual *server*, which clears the flow's packets through its link and also controls the setting up or overbooking of this link.

Assume the packet arrival of the tagged flow is Poisson process with rate λ. The packet length b, measured with the outgoing transmission time (i.e., service time) of the packet, follows a general distribution with mean $\bar{b}$ ($\bar{b} < \infty$), cumulative distribution function (CDF) $B(x)$ and the corresponding Laplace-Stieltjes transform (LST) is denoted by $b^*(s)$.

The Poisson assumption on packet arrivals can be justified by the flow's random behavior of packet generating, which is caused by the new emerging features in the edge networks, such as random sleep scheme, energy-aware traffic shaping and media transcoding, etc. Note that the receipt of a packet at an edge gateway can be considered instantaneous, because the bandwidth of an incoming link to an ingress edge gateway is usually much higher than that of an outgoing link. The Poisson assumption also allows us to obtain closed-form formulas that can still give us enough insights on our scheme's performance.

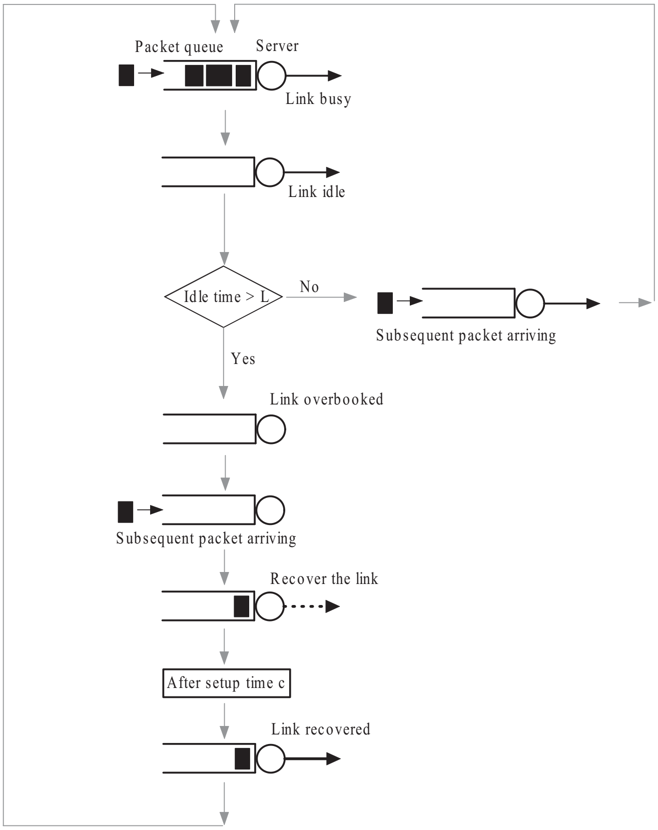

Fig. 2. Demonstration of opportunistic link overbooking scheme on a tagged packet queue.

For a high priority flow, the queued packets are transmitted through its *quasi-dedicated link* in First-In-First-Out (FIFO) order. When the packet queue is emptied, the link becomes idle; once the idle time exceeds the inactivity time L, the server opportunistically overbooks the link to transmit packets for a low priority flow. From the viewpoint of the high priority flow, the server seems to take vacation. And then, when the high priority flow's subsequent packets start arriving, it will take a *setup time* c to recover this link and resume the service. The preemptive task of this link is to guarantee the high priority flow's service. Only when this link is idle, it is overbooked and allowed to temporarily carry a low priority flow's traffic to improve the efficiency, and we thus name it as quasi-dedicated link.

The setup time is involved, because the system has to do some operations to recover this link for the (high priority) tagged queue. For example, in the ATM/MPLS networks, recovering a link needs to restore a series of VPI/VCI configurations along the path and regain the original set of resources [37]. The duration of setup time depends on the features of the specific network, so we suppose c is generally distributed with mean value $\bar{c}$ ($\bar{c} < \infty$), probability density function (PDF) $c(x)$ and LST $c^*(s)$. Because the server takes vacation only after the inactivity time runs out, we get a single server queue whose server is with delayed vacation and setup time.

B. An Equivalent Queueing Model

Under Poisson arrival, the inter-arrival times follow exponential distribution which has the memoryless property. Thus, the idle period I is also exponentially distributed with mean $\bar{I} = 1/\lambda$. Then, with probability $e^{-\lambda L}$, $I > L$, i.e., the first packet in the next busy period encounters a setup time c; with probability $1 - e^{-\lambda L}$, it faces zero setup time.

The setup time can be seen as a prolonged part of the first packet service time. So, we define the *first-packet*, whose length has mean value $\bar{b}_f = \bar{b} + \bar{c}e^{-\lambda L}$, CDF $B_f(x)$ and LST $b_f^*(s)$, where

$$b_f^*(s) = e^{-\lambda L} b^*(s) c^*(s) + (1 - e^{-\lambda L}) b^*(s). \qquad (1)$$

Then the above vacation queue can be equivalently transformed to *a queue with exceptional first packet in each busy period*.

Define the *offered load* $\rho = \lambda\bar{b}$ and the *equivalent load* ρ_e which incorporates the effect of setup time. For an infinite packet queue under OLO scheme, with probability $Q = 1-\rho_e$, an arriving packet sees system empty and is a first-packet. We have $\rho_e = \lambda\bar{\mu}_e$, here $\bar{\mu}_e$ is the *expected packet service time* and

$$\bar{\mu}_e = Q\bar{b}_f + (1 - Q)\bar{b} = \bar{b} + \bar{c}e^{-\lambda L}(1 - \rho_e). \qquad (2)$$

From (2) we can derive

$$\rho_e = \frac{\rho + \lambda\bar{c}e^{-\lambda L}}{1 + \lambda\bar{c}e^{-\lambda L}}. \qquad (3)$$

The correctness of (3) can be verified by the fact that by setting $\bar{c} = 0$ (no setup time introduced) or $L \to \infty$ (the OLO scheme not deployed) result in $\rho_e = \rho$. Given $\bar{c} > 0$, whenever $L < \infty$, $\rho_e > \rho$, i.e., the OLO scheme will introduce excess load to the tagged packet queue and result in service degradation.

C. Opportunistic Link Overbooking Factor

Under the OLO scheme, the overbooked link time from every idle period is $(I-L)^+$, where the operator $(x)^+$ denotes $max(x, 0)$. The mean of the overbooked link time is then

$$\bar{R} = \int_L^\infty (x - L)\lambda e^{-\lambda x} dx = \frac{e^{-\lambda L}}{\lambda}. \qquad (4)$$

Define the *opportunistic overbooking factor* O as

$$O = \frac{\bar{R}}{\bar{I}}(1 - \rho_e) = \frac{\bar{R}}{1/\lambda}(1 - \rho_e) = e^{-\lambda L}(1 - \rho_e). \qquad (5)$$

Note that $\bar{I}$ is the mean link idle period, and from the viewpoint of the high priority flow, the link will be idle with probability $1 - \rho_e$ under OLO scheme.

Thus, in this paper, overbooking means a high priority flow's link can temporarily carry traffic for a low priority flow when the link is idle, while the overbooking factor O means the extra traffic load the link can carry for a low priority flow due to overbooking. The inactivity time L can be used to seek the trade-off between the overbooked link time and the QoS degradation of the high priority flow.

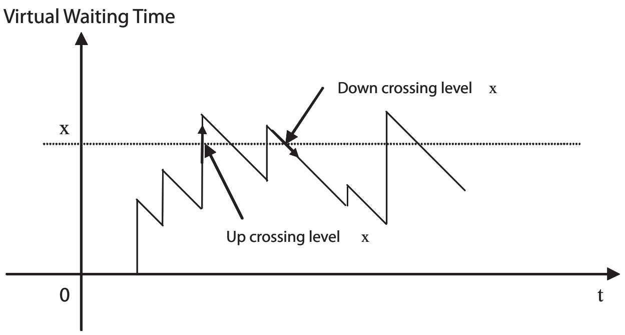

Fig. 3. A realization of virtual waiting time process to illustrate **Lemma 1**.

In the following sections, we will quantitatively analyze the QoS of both delay-sensitive and loss-sensitive flows under the OLO scheme.

III. Delay-Sensitive Flow

Delay-sensitive traffic, such as network games, Voice over IP, has stringent delay and jitter requirements. Thus, we impose the bounded packet delay policy on the packet queue of such traffic, which means only those packets finding their waiting time $\leq T$ can be admitted into the tagged queue. We now analyze the packet delay/jitter and loss probability under OLO scheme based on the equivalent queueing model in Section II.B. The necessary definitions are listed as follows:

For the packet queue with *no delay constraint*:

Expected packet service time	$\bar{\mu}_e$
Probability of system empty	Q
Virtual waiting time (PDF,CDF)	$v(x), V(x)$

For the packet queue with *bounded delay* T:

Expected packet service time	$\bar{\mu}_T$
Probability of system empty	Q_T
Virtual waiting time (PDF,CDF)	$v_T(x), V_T(x)$
Actual waiting time (CDF)	$W_T(x)$

A. Packet Virtual Waiting Time Distribution

As we decide whether or not to admit an arrived packet according to its waiting time, we first must derive the packet waiting time distribution in the tagged queue under OLO scheme. Here, we use the level crossing method [25] to derive two packet virtual waiting time distributions in both the *infinite* packet queue (i.e., with no delay constraint) and the *finite* queue with bounded delay $T, (T \geq 0)$. Then we analyze the packet delay and loss performance based on the *proportional relationship* between these two waiting time distributions.

The *virtual waiting time (VWT)* $\tau(t)$ means the time τ that a packet would have to wait if it arrived at time t. In other words, it is the duration τ needed by the link to clear all backlogged packets in our tagged queue at time t.

The counterpart of virtual waiting time is the *actual waiting time (AWT)*, which occurs often in the finite (blocking) systems, it means the waiting time an admitted packet faces upon arrival until its service is initiated.

1) Level Crossing Method: Considering a single server queue with Poisson arrivals and FIFO service policy, assume that the stationary virtual waiting time process exists and has a unique distribution. Fig. 3 is a sample path of the virtual waiting time process, in which the vertical lines represent new arrivals, who may lead the sample path to upcrossing the level x of *VWT*. On the other hand, the slope lines indicate the decreasing of *VWT* due to the services rendered by server, they may lead the sample path to downcrossing the level x.

Let $H_t(x)$ denotes the number of upcrossings of level x during an arbitrary interval $[0, t]$, then

$$\lim_{t\to\infty} \frac{H_t(x)}{t} = v(x). \tag{6}$$

More precisely, we have the following Lemma [25]:

Lemma 1: In a stationary single server queue with Poisson arrivals and FIFO service policy, the rate of upcrossing a level x of the *VWT* is equal to the rate of downcrossing the level x. In addition, this rate is equal to the probability density function of the *VWT* at x.

2) Proportional Relationship Between VWT Distributions of the Finite and Infinite Packet Queues: The direct consequences of Lemma 1 are the following Lemma 2 and Lemma 3, proved in Appendix I and II, respectively.

Lemma 2: Under OLO scheme, when $\rho_e < 1$, for the packet queue with no delay constraint, the LST of packet *virtual* waiting time PDF $v(x)$ is given by

$$v^*(s) = \frac{\lambda(1-\rho_e)[1-b_f^*(s)]}{s-\lambda[1-b^*(s)]}, \tag{7}$$

where $b_f^*(s)$ and $b^*(s)$ are the LSTs of the first-packet $b_f(x)$ and normal packets $b(x)$, respectively.

Noting that Only if $\rho_e < 1$, the packet queue with no delay constraint can be stable and approach to the equilibrium state. For the corresponding CDF $V(x)$, we have

$$V(0) = Q = 1 - \rho_e. \tag{8}$$

ρ_e is calculated by (3), then from (7) and (8), we can calculate $V(x)$ and derive the *VWT* distribution $V_T(x)$ of the finite queue with bounded delay T according to the following lemma.

Lemma 3: When $\rho_e < 1$, in the interval $[0, T]$, the virtual waiting time distribution of the packet queue with bounded delay T is *proportional* to that of the infinite queue with no delay constraint,

$$v_T(x) = \frac{V_T(0)}{V(0)} v(x) = \frac{V_T(0)}{1-\rho_e} v(x) \quad x \le T, \tag{9}$$

$$V_T(x) = \frac{V_T(0)}{V(0)} V(x) = \frac{V_T(0)}{1-\rho_e} V(x) \quad x \le T. \tag{10}$$

From (7) and (10), we will get the packet delay/jitter and loss rate in the following subsections.

B. Packet Delay and Delay Jitter

For delay-sensitive applications, our interest focuses on the delay of the packets actually admitted into the buffer, i.e., the actual waiting time distribution. Under OLO scheme with bounded delay T, the delay of all admitted packets $\le T$. From Lemma 3, we can get the packet delay and delay jitter as:

Theorem 1: Under OLO scheme, when $\rho_e < 1$, for the packet queue with bounded delay T, the CDF of packet *actual* waiting time is given by

$$W_T(x) = \begin{cases} \dfrac{V(x)}{V(T)} & x < T \\ \\ 1 & x \ge T. \end{cases} \tag{11}$$

The mean packet delay is

$$E[D] = \int_0^T x dW_T(x) = T - \int_0^T \frac{V(x)}{V(T)} dx. \tag{12}$$

The second moment of packet delay is calculated as

$$E[D^2] = \int_0^T x^2 dW_T(x) = \int_0^T x^2 d\frac{V(x)}{V(T)}. \tag{13}$$

And the packet delay jitter is

$$J_T = \sqrt{E[D^2] - E^2[D]}. \tag{14}$$

The proof of Theorem 1 is presented in Appendix III.

C. Packet Loss Rate

While guaranteeing the packet delay, we still concern ourselves with the packet loss performance. Theorem 2 presents the packet loss under OLO scheme with delay constraint T.

Theorem 2: Under OLO scheme, when $\rho_e < 1$, for the packet queue with bounded delay T, the packet loss rate is given by

$$P_T = 1 - \frac{V(T)}{1-\rho_e+\lambda\bar{\mu}_T V(T)}. \tag{15}$$

And the system empty probability is

$$Q_T = \frac{1-\rho_e}{1-\rho_e+\lambda\bar{\mu}_T V(T)}, \tag{16}$$

where $\bar{\mu}_T$ is the expected packet service time,

$$\bar{\mu}_T = \bar{b} + \bar{c}e^{-\lambda L}\frac{1-\rho_e}{V(T)}. \tag{17}$$

The proof of Theorem 2 is presented in Appendix IV.

D. Discussions

We present two examples here to verify the correctness of Theorem 2.

Example 1: If delay constraint $T \to \infty$, no waiting time constraint is imposed on the packets. We then have $V(T)|_{T=\infty} = 1$, and from equation (17) and (2), $\mu_T \to \mu_e$; while in equation (16), $Q_T \to Q = 1 - \rho_e$. Thus, we can calculate packet loss rate from equation (15) as

$$P_T = 1 - \frac{1}{1-\rho_e+\lambda\bar{\mu}_e} = 0, \tag{18}$$

which means no packet loss when no delay constraint is imposed.

Example 2: Another extreme case is when $T = 0$, which means the admitted packet must be served immediately upon its arrival. Actually, we have $V(T)|_{T=0} = Q = 1 - \rho_e$,

$$\mu_T = \bar{b} + \bar{c}e^{-\lambda L} = \bar{b}_f. \tag{19}$$

Now the packet loss becomes

$$P_T = 1 - \frac{1-\rho_e}{1-\rho_e+\lambda\bar{b}_f(1-\rho_e)} = \frac{\lambda\bar{b}_f}{1+\lambda\bar{b}_f}, \tag{20}$$

which is the blocking formula of an $M/G/1/1$ queue [26] with mean packet size $\bar{b}_f$ and CDF $B_f(x)$.

IV. Loss-Sensitive Flow

Loss-sensitive traffic, such as Web browsing/shopping, business transaction data, requires zero packet loss. We thus focus on the loss performance of the tagged queue with buffer size K under OLO scheme, in which only those packets finding the number of waiting packet(s) $< K$ can be admitted. The necessary definitions are listed as follows:

Packet number distribution *at arbitrary time*
$\{P_k\}, k = 0, 1, ...K$
Packet number distribution seen by *arrivals*
$\{\pi_k^a\}, k = 0, 1, ...K$
Packet number distribution seen by *departures*
$\{\pi_k^d\}, k = 0, 1, ...K-1$

As we decide whether or not to admit an arrived packet according to the number of waiting packets ahead of it, we first must derive the packet number distribution in the tagged queue.

A. Packet Number Distribution $\{\pi_k^d\}$ Seen by Departing Packets

1) An Embedded Markov Chain Model: For the packet queue with exceptional first packet in each busy period in Section II.B, we define an embedded Markov chain at the packet departure epochs (after service completion).

Suppose the tagged queue can accommodate at most K packets, including the one in service (transmission). Define n_i the number of packets left behind in the buffer immediately after the transmission of the ith packet. Then, under Poisson arrival, $\{n_i, 0 \le n_i \le K-1; i = 1, 2, ..., \infty\}$ is a Markov chain with transition probability

$$p_{jk} = Prob\{n_i = k | n_{i-1} = j\}. \tag{21}$$

Let

$$\pi_k^d = \lim_{m\to\infty} Prob\{n_{i+m} = k | n_i = j\}, \tag{22}$$

for any $0 \le j \le K-1$, then π_k^d is the steady probability that the departing packets leave k packet(s) in the buffer, i.e., $\{\pi_k^d, 0 \le k \le K-1\}$ is the packet number distribution seen by a departing packet.

Define α_k and β_k as the probabilities that k packet(s) arrived during the service duration of a *first-packet* and a normal packet, respectively. We thus have

$$\alpha_k = \int_0^\infty e^{-\lambda x}\frac{(\lambda x)^k}{k!}dB_f(x) \qquad k = 0, 1, 2, ...; \tag{23}$$

$$\beta_k = \int_0^\infty e^{-\lambda x}\frac{(\lambda x)^k}{k!}dB(x) \qquad k = 0, 1, 2, ... \tag{24}$$

Then

$$p_{0k} = \begin{cases} \alpha_k & 0 \le k \le K-2 \\ \sum_{l=K-1}^\infty \alpha_l & k = K-1, \end{cases} \tag{25}$$

and for $1 \le j \le K-1$

$$p_{jk} = \begin{cases} \beta_{k-j+1} & j-1 \le k \le K-2 \\ \sum_{l=K-j}^\infty \beta_l & k = K-1. \end{cases} \tag{26}$$

Define $\Pi = [\pi_0^d, \pi_1^d, \pi_2^d, ..., \pi_{K-1}^d]$ and

$$\mathbf{P} = \begin{pmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \dots & \dots & \alpha_{K-2} & \sum_{l=K-1}^\infty \alpha_l \\ \beta_0 & \beta_1 & \beta_2 & \dots & \dots & \beta_{K-2} & \sum_{l=K-1}^\infty \beta_l \\ 0 & \beta_0 & \beta_1 & \dots & \dots & \beta_{K-3} & \sum_{l=K-2}^\infty \beta_l \\ 0 & 0 & \beta_0 & \dots & \dots & \beta_{K-4} & \sum_{l=K-3}^\infty \beta_l \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \beta_1 & \sum_{l=2}^\infty \beta_l \\ 0 & 0 & 0 & \dots & \dots & \beta_0 & \sum_{l=1}^\infty \beta_l \end{pmatrix}, \tag{27}$$

From the ergodic theorem of Markov chain [27], Π can be obtained as the solution of

$$\Pi = \Pi\mathbf{P} \tag{28}$$

$$\sum_{k=0}^{K-1} \pi_k^d = 1. \tag{29}$$

2) Recursively Calculation of $\{\pi_k^d\}$: We can recursively calculate $\{\pi_k^d\}$ as follows:

From (27) and (28) we have

$$\pi_k^d = \pi_0^d\alpha_k + \sum_{j=1}^k \pi_j^d\beta_{k-j+1} + \pi_{k+1}^d\beta_0, \quad 0 \le k \le K-2 \tag{30}$$

Let

$$\pi_k{}' = \frac{\pi_k^d}{\pi_0^d}, \quad 0 \le k \le K-1, \tag{31}$$

then

$$\pi'_{k+1} = \frac{1}{\beta_0}(\pi_k{}' - \sum_{j=1}^k \pi_j{}'\beta_{k-j+1} - \alpha_k), 0 \le k \le K-2. \tag{32}$$

These equations form a recursive system with $\pi'_0 = 1$, so that $\{\pi'_k, k = 1, 2, ..., K-1\}$ can be found. From (29) we get

$$\sum_{k=0}^{K-1} \pi'_k = \frac{1}{\pi_0^d}. \tag{33}$$

Thus, we can obtain $\{\pi_k^d, k = 0, 1, ..., K-1\}$, the packet number distribution seen by departing packets, from which we can further derive $\{P_k, 0 \le k \le K\}$, the packet number distribution in the buffer at an arbitrary point of time.

B. Packet Number Distribution $\{P_k\}$, Packet Loss Rate and Mean Delay

Theorem 3: Under OLO scheme, for the packet queue with buffer size K, the packet number distribution is given as

$$P_k = \eta\pi_k^d, \quad k = 0, 1, ..., K-1, \tag{34}$$

where

$$\eta = \frac{1}{\lambda\bar{c}e^{-\lambda L}\pi_0^d + \pi_0^d + \lambda\bar{b}}, \tag{35}$$

and the packet loss rate is

$$P_K = 1 - \frac{1}{\lambda \bar{c} e^{-\lambda L}\pi_0^d + \pi_0^d + \lambda \bar{b}}. \tag{36}$$

The expected packet service time is

$$\bar{\mu}_K = (\bar{b}_f - \bar{b})\pi_0^d + \bar{b}. \tag{37}$$

Proof: Now define $\{\pi_k^a, 0 \le k \le K\}$ as the packet number distribution seen by an arriving packet (whether it can join the buffer or not). Since PASTA (Poisson Arrival See Time Average) [27] holds

$$P_k = \pi_k^a, \quad 0 \le k \le K. \tag{38}$$

And from Burke's Theorem [26]: for any queueing system, in which arrivals and departures occur one by one and that has reached equilibrium state, the packet number distribution seen by the departing packets is the same as the distribution seen by the packets which actually does join the queue. Thus, with probability $1-P_K$, an arriving packet is not blocked and admitted into the buffer, it sees a packet number distribution which is equal to the distribution $\{\pi_k^d\}$ seen by the departures. So

$$P_k = \pi_k^a = (1 - P_K)\pi_k^d, \quad 0 \le k \le K-1. \tag{39}$$

From (39), when $k = 0$, we have

$$P_0 = (1 - P_K)\pi_0^d. \tag{40}$$

While from Little's law [27], we get

$$1 - P_0 = (1 - P_K)\lambda\bar{\mu}_K, \tag{41}$$

where

$$\begin{aligned} \bar{\mu}_K &= \bar{b}_f Prob(\text{packet finding system empty}|\text{packet is admitted}) \\ &\quad + \bar{b}[1 - Prob(\text{packet finding system empty}|\text{packet is admitted})] \\ &= \bar{b}_f \pi_0^d + \bar{b}[1 - \pi_0^d] \\ &= (\bar{b}_f - \bar{b})\pi_0^d + \bar{b}. \end{aligned} \tag{42}$$

Inserting (40) and (42) into equation (41), we can obtain

$$1 = (1 - P_K)[\lambda(\bar{b}_f - \bar{b})\pi_0^d + \pi_0^d + \lambda\bar{b}]. \tag{43}$$

Let $x = 1 - P_K$, then resolving equation (43) yields

$$\eta = \frac{1}{\lambda(\bar{b}_f - \bar{b})\pi_0^d + \pi_0^d + \lambda\bar{b}} = \frac{1}{\lambda\bar{c}e^{-\lambda L}\pi_0^d + \pi_0^d + \lambda\bar{b}} \tag{44}$$

Thus, under OLO scheme, the packet queue with buffer size K suffers packet loss probability

$$P_K = 1 - \frac{1}{\lambda\bar{c}e^{-\lambda L}\pi_0^d + \pi_0^d + \lambda\bar{b}}, \tag{45}$$

and from (39) the packet number distribution is

$$P_k = \eta\pi_k^d, \quad k = 0, 1, ..., K-1. \tag{46}$$

Q.E.D.

Actually, in (36), if we set the setup time $c = 0$, or just let $L \to \infty$ (i.e. the OLO scheme not deployed), P_K becomes $P_K = 1 - \frac{1}{\pi_0^d + \lambda\bar{b}}$, which is the blocking formula of a conventional $M/G/1/K$ queue [26] with packet size CDF $B(x)$.

Then the mean packet number in the buffer is

$$\bar{N} = \sum_{k=0}^{K} kP_k, \tag{47}$$

and from Little's law again,

$$\bar{N} = (1 - P_K)\lambda\bar{W}, \tag{48}$$

so the *mean packet delay* $\bar{W}$ is

$$\bar{W} = \frac{\bar{N}}{(1 - P_K)\lambda} = \frac{\sum_{k=0}^{K} kP_k}{(1 - P_K)\lambda}. \tag{49}$$

C. Discussions

We present two examples here to verify the correctness of Theorem 3.

Example 3: if buffer size $K \to \infty$, no packet is blocked. Actually, when $K \to \infty$, we get $\pi_k^d \to P_k, k = 0, 1, ..., K-1$, i.e., the packet number distribution seen by departures will approach to the distribution seen by any arriving packets. In this infinite packet queue, $P_0 = 1 - \rho_e$, and inserting $\pi_0^d = P_0 = 1 - \rho_e$ into (37), we obtain

$$\bar{\mu}_K = (\bar{b}_f - \bar{b})P_0 + \bar{b} = (1 - \rho_e)\bar{b}_f + \rho_e\bar{b} = \bar{\mu}_e. \tag{50}$$

And then from (44) and (50) the packet loss rate becomes

$$\begin{aligned} P_K &= 1 - \frac{1}{\lambda(\bar{b}_f - \bar{b})P_0 + P_0 + \lambda\bar{b}} \\ &= 1 - \frac{1}{\lambda\mu_e + (1 - \rho_e)} \\ &= 1 - \frac{1}{1} \\ &= 0. \end{aligned} \tag{51}$$

Example 4: when $K = 1$, the tagged queue can only accommodate one packet, so the departing packet finds the buffer empty with probability $\pi_0^d = 1$. From (37), $\bar{\mu}_K = \bar{b}_f$, combining (40) and (41), we obtain

$$P_K = \frac{\lambda\bar{b}_f}{1 + \lambda\bar{b}_f}. \tag{52}$$

Again, we get the blocking formula of an $M/G/1/1$ queue [26] with mean packet size $\bar{b}_f$ and CDF $B_f(x)$, just like the scenario of **Example 2**.

V. Trade-off Between Performance Measures

This section verifies the exactness of our analytical framework and demonstrates the inherent trade-offs between link overbooking factor and flow's QoS measures. We will also show the effectiveness of OLO scheme through further simulations under the more realistic traffic model.

A. Simulation Settings

Simulation results are marked by diamond (◇) or circle (○) in all figures displayed in the sequel. The link capacity is set as $128kbps$, which is a typical granularity for resource management [28].

Link setup time is the duration needed to recover a link, which depends on the features of the specific network. For example, in the ATM/MPLS network, the link refers to a

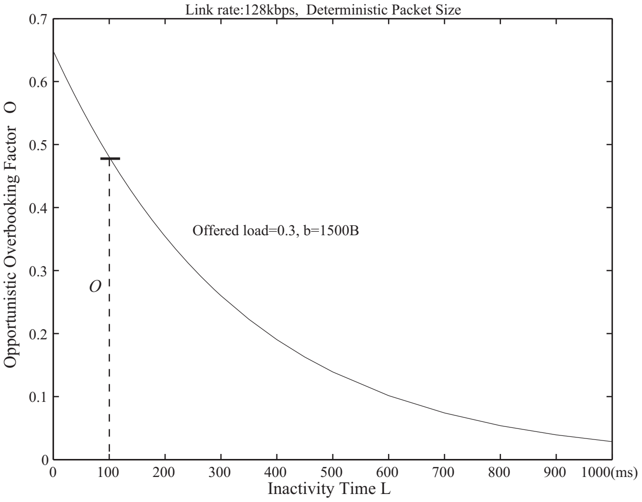

Fig. 4. Opportunistic overbooking factor vs. inactivity time.

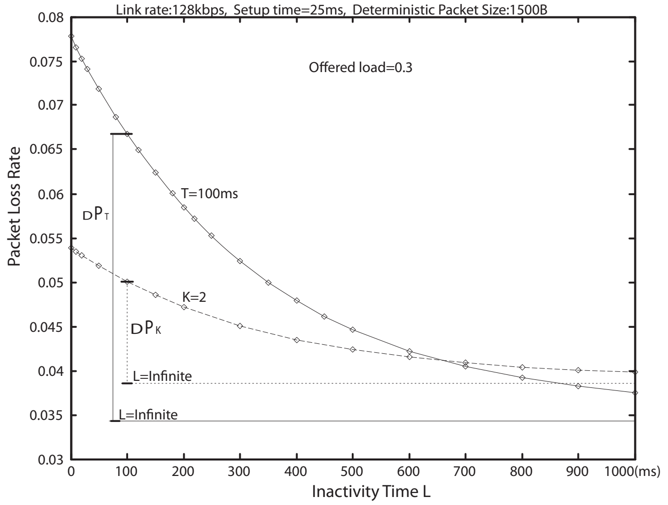

Fig. 5. The impact of inactivity time on packet loss rate: bounded delay vs. finite buffer size.

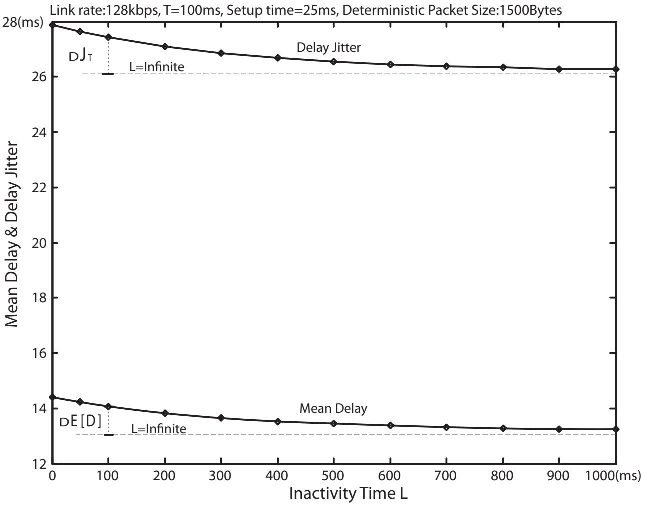

Fig. 6. The impact of inactivity time on mean delay $E[D]$ and jitter J_T: bounded delay case.

virtual circuit, which is identified by a series of VPI/VCI values along the path. Constructing or recovering a circuit needs to set each pair of VPI/VCI on the path and assign the associated resources, which takes tens of milliseconds in a typical ATM-based network [33] [37].

The setup time varies with different network architectures and network states, so we have defined the setup time as a random variable with a general distribution in Section II.A, which enables us to study the impact of variable setup time on our scheme performance. However, there is few concrete result on the statistics of link setup time in various edge networks, some research thus set the setup time as a constant value for convenience sake, e.g., even an out-of-date IP-ATM gateway can use $50ms$ to setup a circuit [29]. Considering the rapid evolution of network hardware, and focusing on the trade-off relationships between overbooking factor and flow's QoS, we set the setup time as a fixed value $25ms$ in our simulations.

As we have introduced, the traffic flow spending much time in silent state results in large and frequent link idle periods, which hence deteriorates the link efficiency. Another concern is that, once the flow becomes active, it may generate large amount of traffic, which often causes buffer overflow or unacceptable packet delay, so it is challenging to guarantee the flow's QoS, especially for the delay/jitter-sensitive flows. In many occasions, the provisioned link capacity has to be large enough to satisfy the QoS requirements. In other words, to effectively reduce the packet loss, enough bandwidth must be allocated to the flow's reserved link to make its offered load relatively small. We thus focus on the light traffic scenario and set the offered load $\rho = 0.3$ in our simulations.

B. Trade-off Between Link Overbooking Factor and QoS

Fig. 4 presents the curve of overbooking factor when $\rho = 0.3$ and the packet size is $1500Bytes$. Inactivity time L directly controls the trade-offs between overbooking factor O and the flow's packet loss, delay/jitter.

Under OLO scheme, as L decreases from $1000ms$ to $0ms$, the factor O, i.e., the extra load the link can carry for a low priority flow due to overbooking, increases from 0.028 to 0.648, which increases link utilization, while degrading high priority flow's QoS. From the solid curve in Fig. 5, the packet loss of delay-sensitive flow will increase from 0.035 to 0.078 when delay constraint $T = 100ms$; while the dashed curve in Fig. 5 shows that loss-sensitive flow's packet loss increases from 0.039 to 0.054 when buffer size $K = 2$ (the buffer can hold at most 2 packets). Furthermore, in Fig. 6, the mean packet delay and delay jitter of delay-sensitive flow increase accordingly when we squeeze the link by shortening the length of L.

For example, if $L = 100ms$, $O = 0.48$, as depicted in Fig. 4. Correspondingly, the QoS degradations are $\Delta P_T, \Delta E[D], \Delta J_T$ (relevant to delay-sensitive flow), and ΔP_K (relevant to loss-sensitive flow), as depicted in Fig. 5 and Fig. 6.

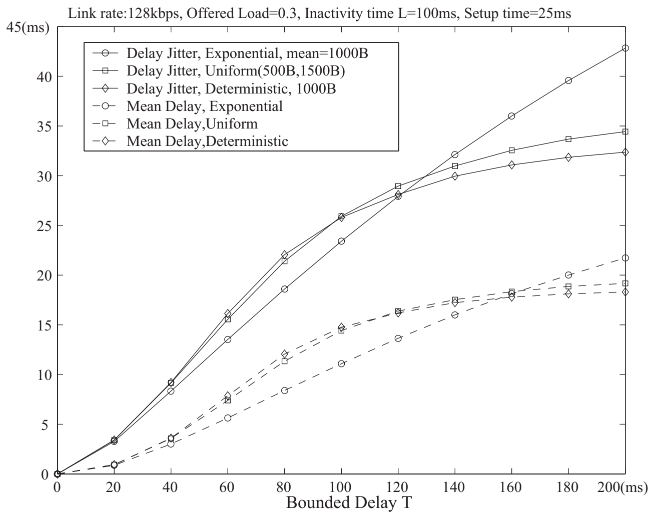

Fig. 7. Mean delay & jitter vs. bounded delay under different packet size distributions.

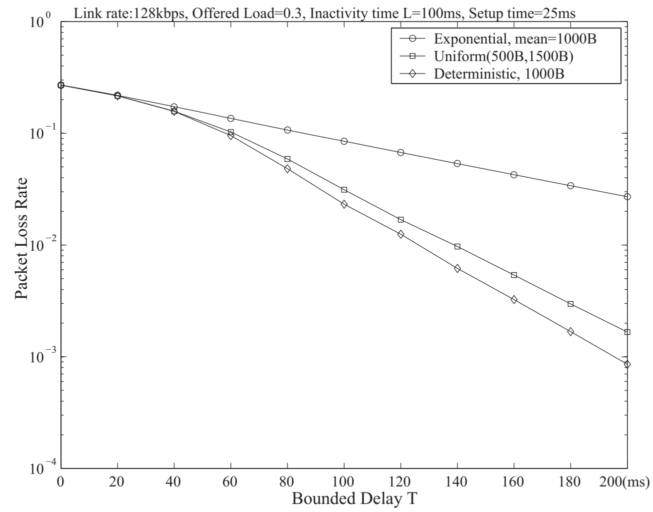

Fig. 8. Packet loss vs. bounded delay under different packet size distributions.

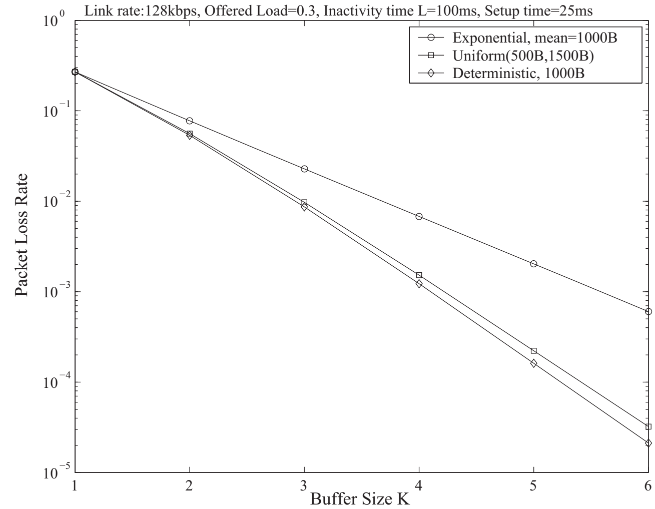

Fig. 9. Packet loss vs. buffer size under different packet size distributions.

C. Impact of Different Traffic Patterns and Packet Admission Constraints

Under OLO scheme, we are also interested in the QoS of high priority flow under different packet size distributions and packet admission policies, which will help us dimension the required transmission resources. Here we consider three probability distributions: *exponential, uniform* and *deterministic*. All the distributions have the same mean packet size $1000Bytes$.

When $L = 100ms$, for delay-sensitive flow, the variations of delay/jitter, packet loss under different delay constraint T and packet size distributions are demonstrated in Fig. 7 and Fig. 8. While for loss-sensitive flow, Fig. 9 presents the curve of packet loss with respect to buffer size K under three packet size distributions.

We see that relaxing the admission constraint (increasing bounded delay T or buffer size K) can effectively decrease the packet loss. Specifically, for delay-sensitive flow, as T increases, the packet delay and delay jitter also become longer.

It is worth noting that, under bounded delay policy, since the delay of all admitted packets are guaranteed, we can trade-off between the allocated link capacity, packet loss and delay/jitter to properly dimension the transmission resources under QoS constraints.

When the above three distributions have the same mean value, the variance of uniform distribution is slightly bigger than that of the deterministic distribution, and exponential distribution has the largest variance which leads to the largest queueing [26], so in Fig. 8 and Fig. 9 the packet losses of the exponential case are much bigger than the other two cases.

D. Further Discussions

In recent years, as core router performance has been dramatically improved, the bottleneck tends to shift from core routers to edge gateways [34]. The OLO scheme is thus aimed at well balancing the link efficiency and high priority flow's QoS for the edge gateways. This relies on the proper setting of link overbooking factor, as depicted in the above subsections. To facilitate the configuration of overbooking factor, we have developed an integrated analytical framework for both delay and loss sensitive flows, where we have applied the Poisson traffic model.

Although Poisson assumption has given us closed-form results and enough insights on the scheme performance, it is known that Poisson model lacks of the ability to capture the diversity of network traffic. Various models have been proposed to emulate all kinds of traffic flows, such as the Markov-modulated rate process (MMRP), Markov-modulated Poisson process (MMPP), long-range dependent (LRD) traffic and self-similar traffic [35][36]. These models are subtle and flexible enough to model versatile traffic sources, but they are also fairly complex in themselves, which makes it difficult to impose them on the analysis of OLO scheme and get the analytical framework.

In order to further demonstrate the performance of OLO scheme under a more realistic traffic scenario, we perform the simulation using MMRP traffic which has been widely used to model various multimedia sources, such as the Voice-IP. An MMRP flow is governed by an M-state Markov

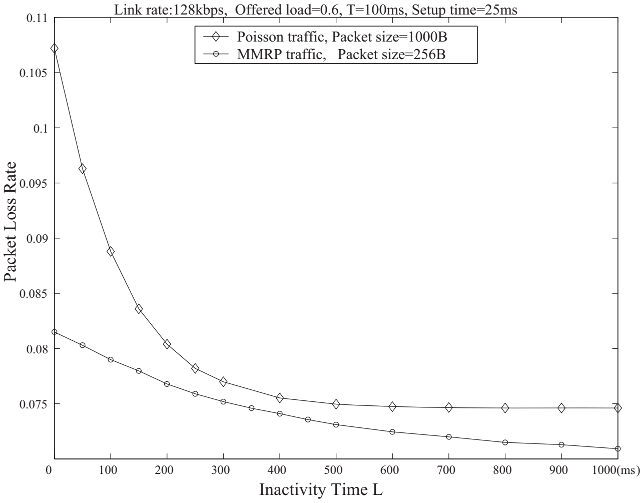

Fig. 10. Packet loss under Poisson and MMRP traffic.

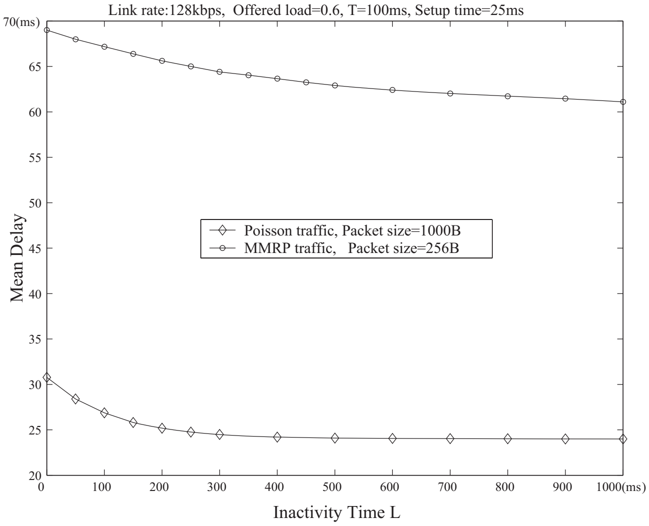

Fig. 11. Mean delay under Poisson and MMRP traffic.

chain with probability transition matrix $Z = \{z_{mn}\}, m, n = 0, 1, ..., M-1$. When the flow is in state m, it moves to state n with probability z_{mn}. The holding time of state m has a general distribution $\Phi_m(x)$. If the flow is in state m, it generates packets at rate r_m [packets/s].

More explicitly, we consider a delay-sensitive flow which has two states ON/OFF. The holding times in ON/OFF states are both exponentially distributed with mean $0.5s$. In ON state, it generates traffic at rate 75 [packets/s]; in OFF state, no traffic is generated. The packet is with a fixed size $256 Bytes$ and the offered load to the link can be computed as $\rho = 0.6$.

Fig. 10 presents the simulated packet loss curve of MMRP flow under OLO scheme with respect to different inactivity time L. For comparison, we also give the loss curve of a Poisson flow with the same offered load 0.6. The simulated mean packet delay of MMRP flow is depicted in Fig. 11.

We can see that, under the same settings, MMRP flow suffers less packet loss than Poisson flow as L varies. Furthermore, packet loss of MMRP is not sensitive to the varying of L, but Poisson flow does. Mean packet delay of MMRP is much bigger than that of Poisson, because MMRP traffic is more bursty than Poisson traffic. Fig. 10 and Fig. 11 imply that, compared with Poisson flow, applying OLO scheme to the MMRP flow can improve link efficiency while introducing smaller impact to flow's QoS, i.e., OLO scheme is more effective on the MMRP flow.

It must be noted that the quasi-dedicated link of a high priority flow is overbooked only when the low priority flows are starving for bandwidth, preventing needless burden on the high priority queue due to recovering the link.

VI. Conclusion and Future Work

In order to well balance the link efficiency and flow's service quality in edge gateways when deploying per-flow QoS, this paper proposed the Opportunistic Link Overbooking (OLO) scheme, which can improve the efficiency of a flow's dedicated link while guaranteeing the service on flow level.

A vacation queueing model has been developed for the scheme, providing an integrated framework to facilitate the setting of overbooking factor and to study the flow's service quality. Further considering the different service requirements of delay-sensitive flow and loss-sensitive flow, we conducted comprehensive queueing analysis under two packet admission policies: 1) bounded packet delay, 2) finite buffer size, which provided different resource dimensioning methods for different QoS profiles.

Our analysis came up with the flow's packet loss, delay and delay jitter under OLO scheme. We also demonstrated the inherent trade-offs between link overbooking factor and flow's QoS under different traffic patterns. The effectiveness of OLO scheme and exactness of its analysis are verified by extensive simulations. Proper link overbooking then can be achieved with the aid of the queueing analysis while still guaranteeing flow's service quality. Another main consequence is that, for delay-sensitive flow, dimensioning its transmission resource under bounded packet delay can provide guaranteed delay performance at the expense of moderate packet loss.

Future endeavors will be aimed at incorporating various traffic models (e.g., MMRP, MMPP) into our analytical framework, which can provide more powerful tool for proper link overbooking. Current simulation settings are mainly focused on verifying the exactness of scheme analysis, although this has shown the effectiveness of OLO scheme, we still need to study the capability of OLO scheme under real network scenarios. Flow's end-to-end behavior under OLO scheme and the impact from the statistical properties of low priority flows are also our future topics.

Appendix A
Proof of Lemma 2

Now we apply Lemma 1 to derive the packet virtual waiting time distribution.

Given $\rho_e < 1$, system is stable, for the packet queue with Poisson arrival of rate λ, the rate of upcrossing level x of virtual waiting time from level 0 is equal to $\lambda(1 - B_f(x))$, and the rate of upcrossing the same level x but starting from level ξ, $0 < \xi \leq x$ is equal to $\lambda(1 - B(x - \xi))$. A direct consequence of Lemma 1 is the following equation

$$v(x) = \lambda(1-B_f(x))V(0)+\lambda \int_0^x (1-B(x-\xi))v(\xi)d\xi, \quad (53)$$

where

$$V(0) = Q = 1 - \rho_e \quad (54)$$

is the probability of system being empty. Taking the derivative of (53), we obtain

$$\frac{dv(x)}{dx} = -\lambda V(0)b_f(x)+\lambda v(x)-\lambda \int_0^x b(x-\xi)v(\xi)d\xi. \quad (55)$$

The Laplace Transform of (55) is

$$sv^*(s)-v(0)+\lambda V(0)b_f^*(s)-\lambda v^*(s)+\lambda b^*(s)v^*(s) = 0. \quad (56)$$

Rearranging the terms and combining with $v(0) = \lambda Q = \lambda(1 - \rho_e)$ yield

$$v^*(s) = \frac{\lambda(1-\rho_e)[1-b_f^*(s)]}{s-\lambda[1-b^*(s)]}, \quad (57)$$

Equation (55) can also be derived from the standard manipulation of the Takacs integro-differential equation [27], but level crossing method is more straightforward.

APPENDIX B
PROOF OF LEMMA 3

Applying **Lemma 1** to the packet queue with bounded delay T based on the same ground for equation (53), we get

$$v_T(x) = \lambda(1 - B_f(x))V_T(0) + \lambda \int_0^x (1 - B(x - \xi))v_T(\xi)d\xi \quad (58)$$

for $x \leq T$. Since equation (58) has the same form as the corresponding equation (53) of the infinite packet queue with no delay constraint. Therefore, in the interval $[0, T]$, the virtual waiting time $v_T(x)$ of the packet queue with bounded delay T is proportional to that $v(x)$ of the infinite queue , and we then obtain

$$v_T(x) = \frac{V_T(0)}{V(0)}v(x) = \frac{V_T(0)}{1-\rho_e}v(x) \qquad x \leq T, \quad (59)$$

$$V_T(x) = \frac{V_T(0)}{V(0)}V(x) = \frac{V_T(0)}{1-\rho_e}V(x) \qquad x \leq T. \quad (60)$$

APPENDIX C
PROOF OF THEOREM 1

First, the equivalent load $\rho_e < 1$ ensures the corresponding infinite queue is stable. Then, for $x \leq T$, the CDF of packet actual waiting time $W_T(x)$ can be derived as

$$\begin{aligned} &W_T(x) \\ &= P(W_T \leq x|\text{the arriving packet is admitted}) \\ &= \frac{P(W_T \leq x, \text{the arriving packet is admitted})}{P(\text{the arriving packet is admitted})} \\ &= \frac{\int_0^x P(W_T \leq x, \text{the arriving packet is admitted}|V_T = y)dV_T(y)}{P(\text{the arriving packet is admitted})} \\ &= \int_0^x \frac{dV_T(y)}{V_T(T)} \\ &= \frac{V_T(x)}{V_T(T)}. \end{aligned} \quad (61)$$

where random variable V_T is the virtual waiting time in the packet queue with bounded delay T. Now combining with (60), we get the delay distribution of admitted packets as

$$W_T(x) = \begin{cases} \dfrac{V(x)}{V(T)} & x < T \\ 1 & x \geq T. \end{cases} \quad (62)$$

APPENDIX D
PROOF OF THEOREM 2

Again, making equivalent load $\rho_e < 1$, under the case of bounded delay, packet loss occurs once the incoming packet finds the virtual waiting time $V_T > T$, which implies packet loss rate

$$P_T = Prob\{V_T > T\} = 1 - V_T(T). \quad (63)$$

Since $V_T(x)$ is related to $V(x)$ in (60), and $V(x)$ is calculated from (57), it remains to derive $V_T(0)$ in order to determine $V_T(x)$. By the definition of virtual waiting time, the steady state probability Q_T that the packet queue with bounded delay T is empty should be equal to the steady state probability that the virtual waiting time $V_T = 0$, that is

$$Q_T = V_T(0). \quad (64)$$

From Little's law [27], we get

$$1 - Q_T = (1 - P_T)\lambda\bar{\mu}_T, \quad (65)$$

where $\bar{\mu}_T$ is the expected packet service time under delay constraint T. Recall that the admitted packets include both $first - packet$s and normal packets, we have

$$\begin{aligned} \bar{\mu}_T &= \bar{b}_f P(\text{packet finding system empty}|\text{packet is admitted}) \\ &\quad + \bar{b}[1 - P(\text{packet finding system empty}|\text{packet is admitted})] \\ &= \bar{b}_f W_T(0) + \bar{b}[1 - W_T(0)] \\ &= (\bar{b} + \bar{c}e^{-\lambda L})\frac{V(0)}{V(T)} + \bar{b}[1 - \frac{V(0)}{V(T)}] \\ &= \bar{b} + \bar{c}e^{-\lambda L}\frac{1-\rho_e}{V(T)}. \end{aligned} \quad (66)$$

Combining (64), (65) and (66) yields

$$V_T(T) = \frac{1 - V_T(0)}{\lambda \bar{\mu}_T}, \tag{67}$$

from (67) and the proportional relationship (60), we get

$$\frac{V_T(0)}{1-\rho_e} V(T) = \frac{1 - V_T(0)}{\lambda \bar{\mu}_T}. \tag{68}$$

Hence

$$Q_T = V_T(0) = \frac{1-\rho_e}{1-\rho_e + \lambda \bar{\mu}_T V(T)}. \tag{69}$$

Inserting (69) back into (60), then for the packet queue with delay constraint T, we obtain the CDF of virtual waiting time as follows

$$V_T(x) = \frac{V(x)}{1-\rho_e + \lambda \bar{\mu}_T V(T)} \quad x \le T, \tag{70}$$

which shows that $V_T(x)$ is a compressed version of $V(x)$ in the interval $[0, T]$. In particular, we have

$$V_T(T) = \frac{V(T)}{1-\rho_e + \lambda \bar{\mu}_T V(T)}. \tag{71}$$

Thus, readily obtained from equation (63), under OLO scheme, the delay-sensitive traffic suffers a packet loss probability as follows

$$P_T = 1 - V_T(T) = 1 - \frac{V(T)}{1-\rho_e + \lambda \bar{\mu}_T V(T)}. \tag{72}$$

References

[1] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture of differentiated services," RFC 2475, Dec. 1998.

[2] J. Ash and W. Lai, "Use of overbooking in Diffserv-aware MPLS traffic engineering," Traffic Engineering Working Group: Internet Draft, June 2003.

[3] F. L. Faucheur, "Protocol extensions for support of Diffserv-aware MPLS traffic engineering," RFC 4124, June 2005.

[4] G. Kesidis, J. Walrand, and C. S. Chang, "Effective bandwidths for multiclass Markov fluids and other ATM sources," *IEEE/ACM Trans. Networking*, vol. 1, pp. 424–428, 1993.

[5] F. P. Kelly, "Notes on effective bandwidths," *Stochastic Networks: Theory and Applications, Royal Statistical Society Lecture Notes Series: 4*, pp. 141-168. Oxford University Press, 1996.

[6] C. Hu, Y. Tang, X. Chen, and B. Liu, "Per-flow queueing by dynamic queue sharing," in *Proc. 26th IEEE Infocom*, May 2007, pp. 1613–1621.

[7] R. Steele, C. C. Lee, and P. Gould, *GSM, CDMAOne and 3G Systems*, 1st edition. Wiley, 2001.

[8] B. Khasnabish, *Implementing Voice over IP*, 1st edition. Wiley-Interscience, 2003.

[9] C. F. Chiasserini and R. R. Rao, "Improving energy saving in wireless systems by using dynamic power management," *IEEE Trans. Wireless Commun.*, vol. 2, no. 5, pp. 1090–1100. Sep. 2003.

[10] C. Poellabauer and K. Schwan, "Energy-aware traffic shaping for wireless real-time applications," in *Proc. IEEE Real-Time Embedded Technology Applications Symposium*, May 2004, pp. 48–55.

[11] C. Poellabauer and K. Schwan, "Energy-aware media transcoding in wireless systems," in *Proc. IEEE Pervasive Computing Commun.*, 2004, pp. 135–144.

[12] I.-H. Huang, C.-S. Lin, C.-S. Chen, and C.-Z. Yang, "Design of a QoS gateway with real-time packet compression," in *Proc. IEEE Tencon*, Oct. 2007, pp. 1–4.

[13] A. Kortebi, L. Muscariello, S. Oueslati, and J. Roberts, "Evaluating the number of active flows in a scheduler realizing fair statistical bandwidth sharing," in *Proc. ACM Sigmetrics 2005*, pp. 217–228.

[14] I. Stoica and H. Zhang, "Providing guaranteed services without per-flow management," *ACM Computer Commun. Review*, vol. 29, no. 4, pp. 81–94, Sep. 1999.

[15] X. Xiao, T. Telkamp, V. Fineberg, C. Chen, and L. Ni, "A practical approach for providing QoS in the Internet backbone," *IEEE Commun. Mag.*, vol. 40, no. 12, pp. 56–62, Dec. 2002.

[16] P. Siripongwutikorn and S. Banerjee, "Per-flow delay performance in traffic aggregates," in *Proc. IEEE Global Telecommun. Conf. (Globecom)*, Nov. 2002, vol. 3, pp. 2634–2638.

[17] J. Liebeherr, S. D. Patek, and A. Burchard, "Statistical per-flow service bounds in a network with aggregate provisioning," in *Proc. 22th IEEE Infocom*, Mar. 2003, pp. 1680–1690.

[18] J. Schmitt, P. Hurley, M. Hollick, and R. Steinmetz, "Per-flow guarantees under class-based priority queueing," in *Proc. IEEE Global Telecommun. Conf. (Globecom)*, Dec. 2003, vol. 7, pp. 4169–4174.

[19] V. Tabatabaee, B. Bhattacharjee, R. La, and M. Shayman, "Differentiated traffic engineering for QoS provisioning," in *Proc. 24th IEEE Infocom*, Mar. 2005.

[20] V. Y. Hnatyshin, "Dynamic bandwidth distribution techniques for scalable per-flow QoS," Ph.D dissertation, University of Delaware, USA, 2003.

[21] Y.-W. Leung, "Dynamic bandwidth allocation for Internet telephony," *Computer Commun.*, vol. 29, no. 18, pp. 3710–3717, Nov. 2006.

[22] J. Elias, F. Martignon, A. Capone, and G. Pujolle, "A new approach to dynamic bandwidth allocation in quality of service networks: performance and bounds," *Computer Networks*, vol. 51, no. 10, pp. 2833–2853, July 2007.

[23] D. Bansat, J. Q. Bao, and W. C. Lee, "QoS-enabled residential gateway architecture," *IEEE Commun. Mag.*, vol. 41, no. 4, pp. 83–89, Apr. 2003.

[24] B. Suter and T. V. Lakshman, "Buffer management schemes for supporting TCP in gigabit routers with per-flow queueing," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 6, pp. 1159–1169, June 1999.

[25] P. H. Brill and J. M. Posner, "Level crossing in point processes applied to queues: single server case," *Operations Research*, vol. 25, July 1977.

[26] J. Medhi, *Stochastic Models in Queueing Theory*. Academic Press, 2003.

[27] L. Kleinrock, *Queueing Systems, Volume I: Theory*. John Wiley & Sons, 1975.

[28] "CCSP Self-Study: Cisco Secure Virtual Private Networks (CSVPN)," 1-58705-145-1. Cisco Press: May 2004.

[29] M. Hassan, R. Sarker, and M. Atiquzzaman, "Modeling IP-ATM gateway using $M/G/1/N$ queue," in *Proc. IEEE Global Telecommun. Conf. (Globecom)*, Nov. 1998, vol. 1, pp. 465–470.

[30] W. Liu, X. Chen, Y. G. Fang, and J. M. Shea, "Courtesy piggybacking: supporting differentiated services in multihop mobile ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 4, pp. 380–393, Oct. 2004.

[31] R. Guimaraes, *et al.*, "Quality of service through bandwidth reservation on multirate ad hoc wireless networks," *Ad Hoc Networks*, vol. 7, no. 2, pp. 388–400, Mar. 2009.

[32] J.-Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer, 2001.

[33] G. R. Ash, "Traffic engineering & QoS methods for IP-, ATM-, TDM-based multiservice networks," Traffic Engineering Working Group, Internet Draft, Mar. 2000.

[34] N. Mihai and G. Vanecek, "New generation of control planes in emerging data networks," *Lecture Notes in Computer Science*, vol. 1653/1999. Springer, 2004.

[35] Q. Ren and H. Kobayashi, "Diffusion approximation modeling for Markov modulated bursty traffic and its applications to bandwidth allocation in ATM networks," *IEEE J. Sel. Areas Commun.*, vol. 16, pp. 679–692, June 1998.

[36] W. B. Gong, Y. Liu, V. Misra, and D. Towsley, "Self-similarity and long range dependence on the Internet: a second look at the evidence, origins and implications," *Computer Networks*, vol. 48, no. 3, pp. 377–399, June 2005.

[37] D. Niehaus and A. Battou, "Performance benchmarking of signaling in ATM networks," *IEEE Commun. Mag.*, vol. 35, no. 8, pp. 134–143, Aug. 1997.

Jianming Liu received the B. Engineering, B. Business Administration, and M. Engineering degrees from the University of Science and Technology of China, in 1999 and 2002, respectively, and the Ph. D degree in Information Engineering from the Chinese University of Hong Kong in 2006. Currently, he is an associate professor at GuiLin University of Electronic Technology, GuangXi, P. R. China. Dr. Liu was also a GCOE research fellow of TOHOKU University, Sendai, Japan, from Dec. 2007 to June 2009. His research interests include wireless sensor networks, optical networks, intelligent control, and the applications of queueing theory.

Xiaohong Jiang received his B.S., M.S., and Ph.D degrees in 1989, 1992, and 1999 respectively, all from Xidian University, Xi'an, China. He is currently an Associate Professor in the Department of Computer Science, Graduate School of Information Science, TOHOKU University, Japan. Before joining TOHOKU University, Dr. Jiang was an assistant professor in the Graduate School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), from Oct. 2001 to Jan. 2005. Dr. Jiang was a JSPS (Japan Society for the Promotion of Science) postdoctoral research fellow at JAIST from Oct. 1999–Oct. 2001. He was a research associate in the Department of Electronics and Electrical Engineering, at the University of Edinburgh from Mar. 1999–Oct. 1999. Dr. Jiang's research interests include optical switching networks, routers, network coding, WDM networks, VoIP, interconnection networks, clock distribution, and fault-tolerant technologies for VLSI/WSI. He has published over 150 referred technical papers in these areas. He is a senior member of IEEE.

Susumu Horiguchi (M'81-SM'95) received the B. Eng, the M. Eng, and PhD degrees from Tohoku University in 1976, 1978, and 1981 respectively. He is currently a Full Professor in the Graduate School of Information Sciences, Tohoku University. He was a visiting scientist at the IBM Thomas J. Watson Research Center from 1986 to 1987. He was also a professor in the Graduate School of Information Science, JAIST (Japan Advanced Institute of Science and Technology). He has been involved in organizing international workshops, symposia, and conferences sponsored by the IEEE, IEICE, IASTED and IPS. He has published over 150 papers technical papers on optical networks, interconnection networks, parallel algorithms, high performance computer architectures, and VLSI/WSI architectures. Prof. Horiguchi is a member of IEICE, IPS, and IASTED.