A Study of the Impact of Network Traffic Pacing from Network and End-User Perspectives

Yan Cai, Y. Sinan Hanay and Tilman Wolf
Department of Electrical and Computer Engineering
University of Massachusetts
Amherst, MA 01003
Email: {ycai, hanay, wolf}@ecs.umass.edu

***Abstract*—As the demand on the Internet bandwidth keeps increasing, all-optical network architectures emerge as a promising solution to high-speed telecommunication networks. However, the performance of optical routers/switches is sensitive to the statistics of Internet traffic due to their limited buffer size, which weakens the optical routers' capability of absorbing Internet traffic surges (known as burstiness), resulting in a high packet drop rate and, hence, lowering the quality of service. As a natural solution to the Internet traffic burstiness, pacing has long been studied and a variety of pacing techniques have been proposed in the literature. Nevertheless, the effect of pacing has not been conclusive. In this paper, we aim to answer the following questions: (1) in what situation pacing is beneficial, (2) does pacing harm short-lived flows, (3) does pacing affect the established fairness? We adopt in our simulation a newly proposed queue length-based pacing technique as a cure to the burstiness. The experimental results show that with appropriate parameter settings, pacing (1) significantly improves the performance of a moderately loaded small-buffer network, (2) brings little negative impact on short-lived flows in a nearly drop-free environment, and (3) is able to maintain the same fairness as achieved on the current Internet. These observations are of significant importance to a potentially wide adoption of small-buffer all-optical networks.**

***Index Terms*—traffic pacing, optical networks, small-buffer networks, performance evaluation**

I. Introduction

As the demand for throughput performance in networks increases, we can expect to see a shift from routers that operate in the electronic domain to routers that operate in the optical domain. While all-optical, packet-switched routers offer higher switching speeds, they also present a considerable challenge from the perspective of network operation: the buffer space of all-optical routers is extremely limited due to practical limitations [1]. Tucker considers this buffer size limitation as one of the factors that make optical routers unattractive [2].

Routers need to employ buffers to store packets due to statistical multiplexing. Buffers are useful when network traffic is bursty. Larger buffers can absorb larger packet bursts at the cost of high queueing delays. In the case of small buffers, packets are more likely to be dropped causing senders that use the Transmission Control Protocol (TCP) to reduce their sending window and thus reduce throughput.

To avoid busrtiness in network traffic and thus to reduce the probability of packet loss, traffic pacing has been developed. Pacing in this context means spacing out packets, so that a group of packets clumped together tightly (i.e., a packet burst) is smoothed by increasing their distances in time domain. Pacing can be thought of as a way of predicting buffer insufficiency that would happen at a future time, and taking countermeasures to prevent packet drops that could occur due to the insufficiency. End-host-based TCP pacing has been the basis of numerous studies on different type of pacing [3]–[5].

Previous studies on this topic have been constrained to simulations and empirical analysis of the data [3], [4], [6], [7]. The results from these studies have lead to unclear or conflicting observations on the benefits and effects of traffic pacing. The specific contribution of our paper is to broadly evaluate the effect of traffic pacing under different network conditions and traffic parameters. We specifically analyze the pacing impact on short-lived flows, and how the pacing performs under different ratios of short-lived and long-lived flows. In addition, we evaluate fairness issues in the context of pacing.

Based on our analysis, we believe researchers can obtain a better understanding to the relationship between the impact of traffic pacing and the underlying network conditions.

The rest of the paper is organized as follows. Section II briefly reviews the pacing technique used in this paper and analyzes its property. Experimental results and their interpretation are presented in Section III. Section IV reviews related works. The paper concludes in Section V.

II. Pacing Technique

We briefly review the pacing technique used in this paper and analyze its property.

A. *Queue Length Based Pacing Technique*

Queue length base pacing (QLBP) is proposed in our prior work [7] and two pacing implementation algorithms in the Open Network Library (ONL) [8] and network simulator 2 (ns2) [9] are presented in [7] and [10], respectively. For completeness, we briefly review the QLBP scheme, followed by some in-depth analyses on its properties. For the details of the algorithms, we refer the reader to [7] and [10].

A QLBP system consists of a pacing queue and a rate controller, as shown in Figure 1 [10]. Such a system has three parameters, $Q_{\max}$ ($\leq Q_{\lim}$), $\mu_{\min}$ and $\mu_{\max}$ ($0 \leq \mu_{\min}, \mu_{\max} \leq C$), where $Q_{\lim}$ is the queue limit of the pacing queue and C the line speed of the outgoing link. $\mu_{\min}$ and

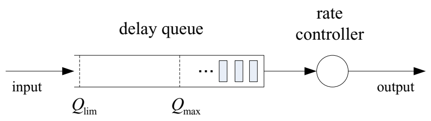


Fig. 1. QLBP System

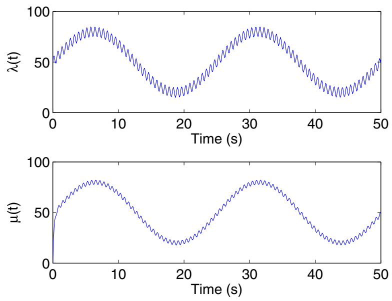


Fig. 2. Pacing effect

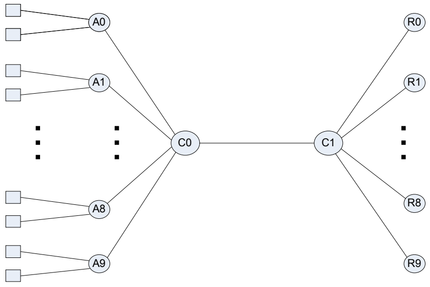


Fig. 3. A dumbbell topology

$\mu_{\max}$ are the minimum and maximum pacing rates. Let $q(t)$ denote the instantaneous queue length at time t, and $\lambda(t)$ and $\mu(t)$ the incoming and pacing rates at time t, respectively. With the given parameters, the pacing rate $\mu(t)$ is determined by

$$\mu(t) = \begin{cases} \frac{\mu_{max}-\mu_{min}}{Q_{max}} q(t) + \mu_{min}, & 0 \leq q(t) \leq Q_{max}, \\ C, & \text{otherwise.} \end{cases} \quad (1)$$

B. Implementation details

To implement this pacing scheme, several technical details need to be considered. A rate controller is used in Figure 1 to dynamically adjust the sending rate (i.e., the pacing rate) of the system. The implementation of a rate controller does not require any additional hardware components, and the pacing effect can be implemented by simply blocking/unblocking the pacing queue. To implement a lower pacing rate, the queue can be blocked for a certain amount time after its sends out a packet. Thus, the QLBP system explicitly enforces a longer "transmission" time for a packet. Notice that any packet is still transmitted in the outgoing link at line speed. Our prototype of QLBP on the NetFPGA reference router [11] demonstrates the feasibility of the QLBP technique. The implemented pacer module does not cost extra processing delays, and the maximum achievable throughput is the same as in the reference router.

C. Non work-conserving property

Clearly, QLBP operates in a non work-conserving fashion, namely, the outgoing link could still be idle when the pacing queue is not empty. This non work-conserving behavior can potentially increase the packet drop probability at the pacer where the QLBP system is deployed. Compared to an ordinary first-in-first-out (FIFO) queue, an QLBP queue has a weaker capacity of absorbing traffic surges because of the combinational effect of a lower sending rate and a shorter available buffer space. To make it more concrete, we compare a FIFO queue and an QLBP queue of the same size in the following scenario. Suppose that both the FIFO queue and the QLBP queue are fed with the same input process $\lambda(t)$ (> 0). For the FIFO queue, $q(t)$ is always zero whenever $\lambda(t) < C$, whereas, for the QLBP queue, $q(t)$ is larger than zero for all time. The sending rate, $\mu(t)$, of the QLBP queue is smaller than that of the FIFO queue, i.e., the line speed C. Assume that at time t_0, $\lambda(t)$ jumps to a constant rate λ_0 ($> C$) for the first time and lasts for $\frac{Q_{\lim}}{\lambda_0 - C}$. Thus, at time $t_0 + \frac{Q_{\lim}}{\lambda_0 - C}$, the buffer size of the FIFO queue is just equal to $Q_{\lim}$, and there is no drop in the FIFO queue between t_0 to $t_0 + \frac{Q_{\lim}}{\lambda_0 - C}$. However, such a surge can cause packet drops in the QLBP queue because of a shorter available buffer size, i.e., $Q_{\lim} - q_0$, where q_0 (> 0) is the length of the QLBP queue at time t_0.

D. Analysis of Paced Traffic in Frequency Domain

Pacing can reduce the burstiness of the incoming traffic, and hence lower the packet drop probability at downstream routers. We carry out the following calculation to demonstrate the burst-reducing effect at different frequencies.

It is pointed in [10] that under the condition that $\mu_{\min} = 0$, $\mu_{\max} = C$, $Q_{\max} < Q_{\lim}$ and $0 < \lambda(t) < C$, the relationship between the incoming rate $\lambda(t)$ and the pacing rate $\mu(t)$ can be described by

$$\frac{d}{dt}\mu(t) = -\alpha\mu(t) + \alpha\lambda(t), \quad (2)$$

where $\alpha = \frac{\mu_{\max} - \mu_{\min}}{Q_{\max}}$, known as the pacing coefficient. Assuming $\mu(0) = 0$ and taking Laplace transform on both sides of Eq. (2), we have

$$U(s) = \frac{\alpha}{s + \alpha}\Lambda(s), \quad (3)$$

where $U(s)$ and $\Lambda(s)$ are the Laplace transforms of $\mu(t)$ and $\lambda(t)$, respectively. Thus, given an input traffic signal $\lambda(t)$, the frequency information of $\mu(t)$ can be completely determined using Eq. (3).

In what follows, we use a toy traffic model to demonstrate the pacing effect of QLBP on different frequencies of an underlying network traffic. Here the incoming traffic $\lambda(t)$ is modeled by

$$\lambda(t) = c_0 + h_1 \sin(\omega_1 t) + h_2 \sin(\omega_2 t), \quad (4)$$

where $c, h_1, h_2, \omega_1, \omega_2$ satisfy the constraints below,

$$\begin{cases} c_0 > h_1 \gg h_2, \\ c_0 + h_1 + h_2 < C, \\ \omega_1 \ll \omega_2. \end{cases} \quad (5)$$

The traffic model (4) is motivated by an observation that the Internet traffic oscillates at large time scales, namely, tens of minutes (see Fig. 1-(a) in [12]). In the meantime, the Internet traffic changes quite dramatically at small time scales, for instance, lower than hundreds of milliseconds [13]. Motivated by such observations, we use the terms $h_1 \sin(\omega_1 t)$ and $h_2 \sin(\omega_2 t)$ to model traffic burst at the two significantly different time scales ω_1 and ω_2. The top constraint in (5) guarantees that $\lambda(t) > 0$. The middle inequality ensures $\lambda(t) < C$. The last one reflects the fact that the two traffic components are far apart from each other in the frequency domain.

Notice here that by no means we intend to make a claim that the Internet traffic exhibits any kind of periodicity as described by Eq. (4). Our incentive for adopting such a model is to be able to find a model for the traffic signal that contains the components at different frequencies. Eq. (4) is an extremely simplified model in which the power of the incoming traffic signal is all concentrated at two frequencies ω_1 and ω_2.

For the incoming traffic $\lambda(t)$ given by Eq. (4), its Laplace transform is given by

$$\Lambda(s) = \frac{c_0}{s} + \frac{h_1\omega_1}{s^2+\omega_1^2} + \frac{h_2\omega_2}{s^2+\omega_2^2}. \tag{6}$$

Substituting it into Eq. (3), we obtain

$$U(s) = \frac{\alpha}{s+\alpha}\left(\frac{c_0}{s} + \frac{h_1\omega_1}{s^2+\omega_1^2} + \frac{h_2\omega_2}{s^2+\omega_2^2}\right). \tag{7}$$

Taking inverse Laplace transform for the equation above, we have

$$\begin{aligned}\mu(t) &= c_0(u(t) - e^{-\alpha t}) \\ &+ \frac{h_1\alpha\omega_1}{\omega_1^2+\alpha^2}e^{-\alpha t} + \frac{h_1}{\sqrt{(\omega_1/\alpha)^2+1}}\sin(\omega_1 t - \gamma_1) \\ &+ \frac{h_2\alpha\omega_2}{\omega_2^2+\alpha^2}e^{-\alpha t} + \frac{h_2}{\sqrt{(\omega_2/\alpha)^2+1}}\sin(\omega_2 t - \gamma_2),\end{aligned} \tag{8}$$

where γ_1, γ_2 are given by

$$\gamma_i = \arcsin\left(\frac{1}{\sqrt{1+(\alpha/\omega_i)^2}}\right), \text{ for } i = 1, 2.$$

For $t \gg 0$, we have

$$\begin{aligned}\mu(t) = c_0 u(t) &+ \frac{h_1}{\sqrt{(\omega_1/\alpha)^2+1}}\sin(\omega_1 t - \gamma_1) \\ &+ \frac{h_2}{\sqrt{(\omega_2/\alpha)^2+1}}\sin(\omega_2 t - \gamma_2).\end{aligned} \tag{9}$$

For an α ($\omega_1 \ll \alpha < \omega_2$), $\omega_1/\alpha \approx 0$ and $\omega_2/\alpha > 1$, and, therefore, $\frac{h_1}{\sqrt{(\omega_1/\alpha)^2+1}} \approx h_1$ and $\frac{h_2}{\sqrt{(\omega_2/\alpha)^2+1}} < h_2$.

Figure 2 shows an example of the pacing effect. The upper figure represents the trajectory of $\lambda(t)$ (given by Eq. (4)) and the lower figure the trajectory of $\mu(t)$ (given by Eq. (8)). In this case, the parameters are set as follows: $c_0 = 50, h_1 = 30, h_2 = 5, \omega_1 = 0.2512, \omega_2 = 10, \alpha = 5$. It can be seen that the high frequency oscillation in $\lambda(t)$ is effectively suppressed, resulting in a smoothed $\mu(t)$. In the meantime, the lower frequency component is rarely affected.

In practice, there is no evidence that the Internet traffic exhibits any periodicity at small time scales. However, the above analysis can be extended to the real Internet traffic in the following way. Assume that α is chosen to curb burstiness at time scales of the order of 10ms. We take an input traffic long of thousands of seconds. By doing so, we aim to eliminate the impact of boundary conditions. We make up a periodic signal by cascading duplicate copies of the input traffic together. Such a periodic signal can be expressed as a Fourier series, and the rest operations of Fourier transform and inverse Fourier transform can be carried through. As a result, for these components whose frequency ω is larger than α, their amplitudes are reduced by a factor of $1/\sqrt{1+(\omega/\alpha)^2}$ and for these components whose frequencies are far smaller than α, their amplitudes almost keep unchanged. Notice that since the coefficient $1/\sqrt{1+(\omega/\alpha)^2}$ is always smaller than 1, the amplitudes at all frequencies are never amplified.

III. Experimental Evaluation

In this section we evaluate the pacing impact on various network applications via simulation.

A. System Metrics

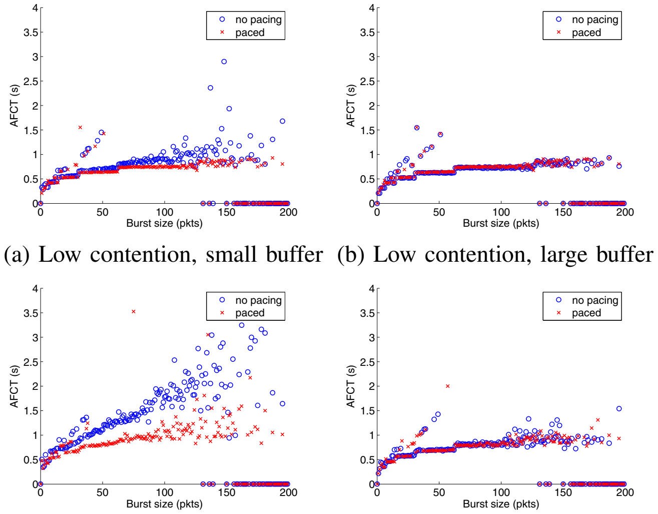
We first introduce the system metrics used in simulations. The first metric is a so-called average flow completion time (AFCT), the use of which is justified in [14]. Flow completion time is different from link utilization, throughput and fairness; it reflects how promptly network-based applications, such as web-surfing and instant messages, can respond to actions of end-users. We use it as a metric to quantitatively characterize the pacing impact on short-lived flows. The second is throughput, which is used for long-lived flows. This metric determines the underlying network's capacity to deliver real-time services, such as video on demand or teleconference. The third one is TCP fairness. A great concern for TCP pacing is the disadvantage of paced TCP flows in competing with non-paced TCP flows [3]. We use it to evaluate how the bottleneck bandwidth is shared between paced and non-paced aggregate traffics. We use the Jain's fairness index [15] as a measure of fairness. It is defined as

$$J(\vec{x}) = \frac{(\sum_{i=1}^n x_i)^2}{n\sum_{i=1}^n x_i^2}, \tag{10}$$

where $\vec{x} = (x_1, \cdots, x_n)$ is the throughput vector of n flows.

B. Experimental Setup

The entire set of simulations is run in ns2. Figure 3 shows the dumbbell topology used in the simulations. All links have a capacity of 100Mbps. The transmission delays of all links are set such that the average round trip time between a sender/receiver pair is 100ms. The bottleneck link is between C0 and C1. Ten access routers A0 to A9 are attached to C0 while ten receivers R0 to R9 to C1. Each access router is connected to two senders. Buffer sizes of all links other



(a) Low contention, small buffer (b) Low contention, large buffer

(c) High contention, small buffer (d) High contention, large buffer

Fig. 4. Impact of pacing on average flow completion time

than the bottleneck link are set 2000pkts. The buffer size of the bottleneck link varies from 0 to 100pkts. This way, we ensure that all packet drops only occur at the bottleneck link, rather than anywhere else. In the case that pacing is enabled, $\mu_{\max} = 100$Mbps, $\mu_{\min} = 1$Mbps, $Q_{\max} = 40$pkts.

The traffic model is a mix of long-lived and short-lived TCP flows. Each sender sends a number of long-lived TCP flows and a number of short-lived TCP flows. The TCP packet size is 1000Bytes. The maximum congestion and receiver windows of a long-lived TCP flow are 32pkts and 64pkts, respectively. The maximum congestion and receiver windows of a short-lived TCP flow are both 64pkts. A long-lived TCP flow is just an FTP application that can send data persistently during a run while a short-lived flow intermittently sends a bunch of packets, called a packet burst. The average burst size is 20pkts and the average interval between two consecutive bursts is 2.25seconds. With such settings, the maximum throughput of a long-lived TCP flow is 2.5Mbps, and the average throughput of a short-lived flow is 71Kbps (=20pkts*8Kbits/pkt / 2.25s), and a short-lived flow spends 99.8% of its active time in slow-start phase. The active time of a short-lived flow means the time during which it either sends packets or waits for acknowledgements.

All short-lived and long-lived flows start randomly in the first 100 seconds. Each run lasts 2000s and the transient state ends at 200s. All measures are the steady state measures that are calculated over an interval [200s, 2000s].

C. Experimental Results

1) Average Flow Completion Time of Short-Lived Flows: We evaluate the impact of pacing on the performance of short flows in terms of average flow completion time (AFCT). The transmission delays are set as follows. The link delays between senders to access routers and C1 to receivers are uniformly distributed in [1ms, 10ms]. The link delays between access routers are all 10ms and the link delay between C0 to C1 is 30ms. Such settings satisfy that the average round trip transmission delay between a pair of sender and receiver is 100ms. The number of long-lived flows varies is 0 or 40. The number of short-lived flows is 100. The buffer size of the bottleneck link is 5pkts or 100pkts. Parameter settings are given in Table I.

TABLE I
PARAMETER SETTINGS IN FIGURE 4

Subfigure	# of l.f.	Buffer Size (pkts)	# of s.f.
(a)	0	5	100
(b)	0	100	100
(c)	40	5	100
(d)	40	100	100

Figure 4 shows the impact of pacing on AFCT under different conditions. In each subfigure, the x-axis is the burst size and the y-axis is the AFCT, where the AFCT of burst size s_0 is defined as the average completion time of all bursts long of s_0 packets. The dotted blue curves and the dashed red curves represent non-paced and paced short-lived flows, respectively. From them, we make the following observations.

First, from Figures 4(a) and 4(c), we can see pacing is beneficial when buffer size is small. In the case of Figure 4(a), although there are no long-lived TCP flows competing with short-lived flows, short-lived flows still experience packet drops occasionally due to a combinational effect of TCP burstiness and limited buffer size. In the case of Figure 4(c), existence of long-lived flows makes the bottleneck link more congested, and as a result, short-lived flows experience drops more frequently. Due to pacing effect, TCP burstiness is significantly reduced and AFCT is shortened (comparing non-paced and paced curves in Figures 4(a) and 4(c)).

Second, when the buffer size is so large that TCP burstiness can be absorbed significantly, AFCTs of paced and unpaced short-lived flows are quite similar, as shown in Figures 4(b) and 4(d), indicating there is little impact of pacing on short-lived flows. This is because the pacing delay is much small than the RTT, which mainly determines the AFCT of a short-lived flow in the context of additive-increase multiplicative-decrease TCP [16].

2) Throughput of Long-Lived Flows: Now we turn to the impact of pacing on the performance of long-lived flows. Note that a similar study has been conducted in [10]. Our following work differs from [10] in that the underlying traffic in our simulation is a mix of short-lived and long-lived flows while in [10] there is no use of short-lived flows.

Figure 5 and 6 show how QLBP affects long-lived flows in terms of throughput. In Figure 5 and 6, x-axis is the buffer size at the bottleneck link while y-axis is the normalized total throughput of all long-lived flows, which is defined as the radio of the total throughput to the capacity of the bottleneck link. In Figure 5, there are 20 long-lived flows. Since the maximum throughput of each long-lived flows is 2.5Mpbs, the total maximum throughput is roughly 50Mbps, representing a case of the light traffic load. In Figure 6 there are 40 long-lived flows, representing a case of moderate traffic load [17]. Note that without contention, 100 and 900 short-lived

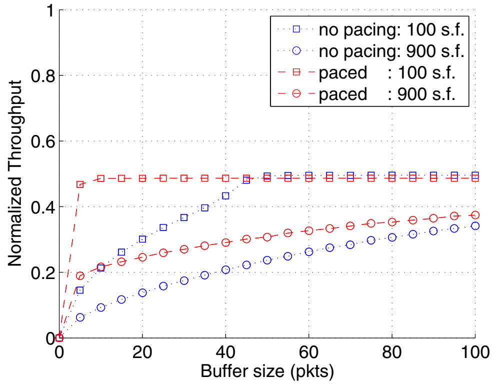


Fig. 5. Light load: 20 long-lived flows

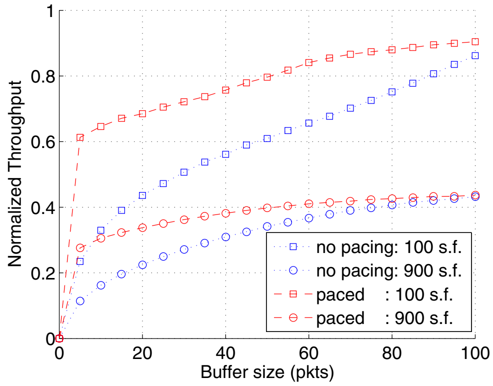


Fig. 6. Light load: 40 long-lived flows

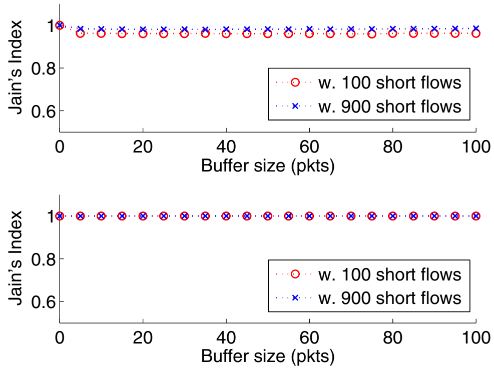


Fig. 7. Fairness between paced and non-paced long-lived flows

flows contribute about 7.1Mbps and 64Mbps of traffic to the bottleneck link.

We make the following observations.

First, without pacing, long-lived TCP flows can't fully utilize the shared bandwidth when the buffer size at the bottleneck link is small. For example, with a buffer size smaller than 40pkts and existence of 100 short-lived flows in Figure 5 (see the dotted curve with square mark, the total throughput of all long-lived flows is smaller than its theoretical maximum, i.e., 50Mbps. A similar trend exists for other cases with different degrees of regression.

Second, pacing can improve the throughput of long-lived flows for small buffers. In Figure 5 and 6 we compare all cases with and without pacing, the improvement on throughput is shown by the gap between the pair of corresponding performance curves. For example, for a buffer size of 20pkts, pacing can improve the total throughput by 50% (see curves of 900 short-lived flows in Figure 6) to nearly 100% (see curves of 900 short-lived flows in Figure 5).

Third, as the buffer size at the bottleneck link gets bigger, the benefit of pacing diminishes. This is because the impact of TCP burstiness that is what the pacing targets is weakened by the increased buffer size.

We conduct more experiments with different parameters. They all show the same trend.

3) TCP Fairness: We now study the fairness between non-paced and paced traffic. Each sender node establishes 2 long-lived flows and 5/45 short-lived flows. In total, there are 40 long-term flows and 100/900 flows. The average inter-burst time is set 10seconds. Rest of the settings are still the same as described before. We disable QLBP at access router A5 to A9 in Figure 3. Thus, the first half of the total traffic flows are paced, and the second half not.

The upper subfigure in Figure 7 shows the fairness $J(\vec{x})$ versus the various buffer size with different numbers of short-lived flows. In this case, the throughput vector $\vec{x}$ has 40 elements: the first 20 represent the throughput of paced long-lived flows while the second 20 the throughput of non-paced long-lived flows. We can see the Jain's index $J(\vec{x})$ is quite close to 1, indicating the TCP fairness is established between these paced and non-paced flows.

The lower subfigure in Figure 7 also shows the fairness $J(\vec{x})$ versus the various buffer size with different numbers of short-lived flows, where $\vec{x} = (x_1, x_2)$ and x_1 and x_2 are the aggregate throughput of the first and second 20 long-lived flows, respectively. We use such a $\vec{x}$ to study if paced traffics are treated differently from non-paced traffic at the bottleneck link. From the figure we can see no matter how many short-lived flows exist and how big the bottleneck buffer size is, the bottleneck link give paced and non-paced traffic the same forwarding priority, indicating that non-paced traffic has no disadvantage in competing with non-paced traffic.

IV. Related Work

IP QoS mechanisms are used to ensure that a network can provide certain levels of services that meet the requirements specified in the service level agreement (SLA). Burstiness, as a characteristic of IP traffic, has a close relationship with the metrics defined in the SLA [18].

As a natural solution to the burstiness of TCP traffic, TCP pacing has been studied for a while. TCP pacing was initially proposed by Zhang et al. in [19] to correct for the acknowledgement compression due to cross traffic. Since then, many researches have followed, proposing different usages of pacing in TCP for different purposes, such as compensation of ACK-clocking in slow start [20], [21] and fast recovery after a packet loss [22], reducing burstiness in asymmetric networks [23]. The conclusions on the impact of TCP pacing on network performance have been controversial. On one hand, empirical studies conducted in [3] indicate that although TCP pacing can improve throughput and fairness in some situations, it degrades the performance of TCP in general. On the other hand, TCP pacing is necessary for TCP sessions to achieve the high link utilization in case of the bottleneck equipped with small buffers [17], [24]–[27]. Notice that all high performance results are achieved only when TCP sessions are paced by either some rate-control mechanism, or access links with capacities much slower than the bottleneck link.

In addition to TCP pacing, there have been other proposals for resolving packet drops in small buffer networks [28]–[31]. The work by Sivaraman et al. [28] stems from previous works on traffic conditioners for video transmission, called traffic conditioning *off-line*. They proposed an on-line version of traffic conditioner based on this traffic conditioning *off-line*. These approaches in [29]–[31] rely on the global network-wide coordinated scheduling.

V. Conclusions

In this paper we systematically study the impact of pacing on various network traffic conditions in the context of QLBP. The analysis of frequency domain shows that QLBP can suppress the burstiness of network traffic by reducing amplitudes of signals at high frequencies. Through extensive simulations, we answer the three questions raised at the beginning of the paper. First, pacing is beneficial for a small buffer network where the performance of the underlying network degrades significantly due to the burstiness of network traffic. Second, pacing with appropriate settings does not harm short-lived flows in a nearly drop-free environment. Third, pacing has little effect on the established TCP fairness. Since the potential all-optical networks are of exactly small buffer, we believe it is necessary to involve pacing in the future all-optical network architectures. Our studies in this paper clarify the potential impact of pacing in such a network, providing a better vision to the adoption of pacing in future networks.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. CNS-0721790.

References

[1] P. Bernasconi, J. Gripp, D. Neilson, J. Simsarian, D. Stiliadis, A. Varma, and M. Zirngibl, "Architecture of an integrated router interconnected spectrally (iris)," in *High Performance Switching and Routing, 2006 Workshop on*, 2006, p. 8 pp.

[2] R. S. Tucker, "The role of optics and electronics in high-capacity routers," *Lightwave Technology, Journal of*, vol. 24, no. 12, pp. 4655 –4673, 2006.

[3] A. Aggarwal, S. Savage, and T. Anderson, "Understanding the performance of TCP pacing," in *Proc. of IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar. 2000, pp. 1157–1165.

[4] S. M. ElRakabawy, A. Klemm, and C. Lindemann, "Tcp with adaptive pacing for multihop wireless networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, ser. MobiHoc '05. New York, NY, USA: ACM, 2005, pp. 288–299. [Online]. Available: http://doi.acm.org/10.1145/1062689.1062726

[5] G. Hasegawa, T. Tomioka, K. Tada, and M. Murata, "Simulation studies on router buffer sizing for short-lived and pacing tcp flows," *Computer Communications*, vol. 31, no. 16, pp. 3789 – 3798, 2008, performance Evaluation of Communication Networks (SPECTS 2007).

[6] O. Alparslan, S. Arakawa, and M. Murata, "Node pacing for optical packet switching," *IEICE technical report*, vol. 107, no. 544, pp. 17–20, 20080306. [Online]. Available: http://ci.nii.ac.jp/naid/110006884727/en/

[7] Y. Cai, S. Hanay, and T. Wolf, "Practical packet pacing in small-buffer networks," in *ICC '09*, Dresden, Germany, Jun. 2009.

[8] J. DeHart, F. Kuhns, J. Parwatikar, J. Turner, C. Wiseman, and K. Wong, "The open network laboratory: a resource for networking research and education," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 5, pp. 75–78, Oct. 2005.

[9] The network simulator - ns-2, http://www.isi.edu/nsnam/ns/.

[10] Y. Cai, B. Jiang, T. Wolf, and W. Gong, "A practical on-line pacing scheme at edges of small buffer networks," in *Proc. of IEEE INFOCOM 2010*, San Diego, CA, Mar. 2010.

[11] "Netfpga," http://www.netfpga.org/.

[12] T. Wolf, Y. Cai, P. A. Kelly, and W. Gong, "Stochastic sampling for internet traffic measurement," in *Proc. of 10th IEEE Global Internet Symposium*, Anchorage, AK, May 2007.

[13] A. Feldmann, A. Gilbert, P. Huang, and W. Willinger, "Dynamics of ip traffic: A study of the role of variability and the impact of control," in *Proc. of ACM SIGCOMM '99*, Aug. 1999, pp. 301–313.

[14] N. Dukkipati and N. McKeown, "Why flow-completion time is the right metric for congestion control," *SIGCOMM Comput. Commun. Rev.*, vol. 36, pp. 59–62, January 2006. [Online]. Available: http://doi.acm.org/10.1145/1111322.1111336

[15] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," DEC, Technical Report TR301, 1994.

[16] A. Lakshmikantha, R. Srikant, N. Dukkipati, N. McKeown, and C. Beck, "Buffer sizing results for RCP congestion control under connection arrivals and departures," *SIGCOMM Computer Communication Review*, vol. 39, no. 1, Jan. 2009.

[17] M. Enachescu, Y. Ganjali, A. Goel, N. McKeown, and T. Roughgarden, "Routers with very small buffers," in *Proc. of IEEE INFOCOM 06*, Spain, Apr. 2006, pp. 1–11.

[18] J. W. Evans and C. Filsfils, *Deploying IP and MPLS QoS for Multiservice Networks: Theory & Practice*. Morgan Kaufmann, 2007.

[19] L. Zhang, S. Shenker, and D. D. Clark, "Observations on the dynamics of a congestion control algorithm: the effects of two-way traffic," in *ACM Computer Communication Review*, 1991, pp. 133–147.

[20] M. Aron and P. Druschel, "TCP: Improving startup dynamics by adaptive timers and congestion control," Rice University, Technical Report TR98-318, 1998.

[21] V. N. Padmanabhan and R. H. Katz, "TCP Fast Start: A technique for speeding up web transfers," in *Proc. of IEEE GLOBECOMM*, Sydney, Australia, Nov. 1998.

[22] J. Hoe, "Start-up dynamics of TCP's congestion control and avoidance schemes," Masterthesis, MIT, Jun. 1995.

[23] H. Balakrishnan, V. N. Padmanabhan, and R. H. Katz, "The effects of asymmetry on tcp performance," in *Mobicom 1997*, Sep. 1997.

[24] D. Wischik and N. McKeown, "Part I: Buffer sizes for core routers," *ACM SIGCOMM Comput Commun Rev*, pp. 75–78, Jul. 2005.

[25] G. Raina, D. Towsely, and D. Wischik, "Part II: Control theory for buffer sizing," *ACM SIGCOMM Comput Commun Rev*, pp. 79–82, Jul. 2005.

[26] Y. Gu, D. Towsely, C. V. Hollot, and H. Zhang, "Congestion control for small buffer high speed networks," in *Proc. of IEEE INFOCOM 07*, Anchorage, Alaska, May 2007, pp. 1037–1045.

[27] A. Lakshmikantha, R. Srikant, and C. Beck, "Impact of file arrivals and departures on buffer sizing in core routers," in *Proc. of IEEE INFOCOM 08*, Pheonix, AZ, Apr. 2008, pp. 86–90.

[28] V. Sivaraman, H. Elgindy, D. Moreland, and D. Ostry, "Packet pacing in short buffer optical packet switched networks," in *Proc. of IEEE INFOCOM 06*, Spain, Apr. 2006.

[29] J. Naor, A. Rosen, and G. Scalosub, "Online time-constrained scheduling in linear networks," in *Proc. of IEEE INFOCOM 05*, Miami, FL, Mar. 2005.

[30] M. Adler, S. Khanna, R. Rajaraman, and A. Rosen, "Time-constrained scheduling of weighted packets on trees and meshes," *Algorithmica*, vol. 36, no. 2, pp. 123–152, 2003.

[31] M. Adler, A. L. Rosenberg, R. K. Sitaram, and W. Unger, "Scheduling time-constrained communication in linear networks," *Theoretical Comp. Sc*, vol. 35, no. 6, pp. 559–623, 2002.