RED-Worcester – Traffic Sensitive Active Queue Management

Vishal Phirke, Mark Claypool, Robert Kinicki
CS Department, Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609, USA
{vish|claypool|rek}@cs.wpi.edu

1 Introduction

The Internet today carries traffic for applications with a wide range of delay and throughput requirements. Traditional applications such as FTP and E-mail can tolerate high delays due to long router queues in exchange for high throughput. Emerging applications such as IP telephony, video conferencing and networked games, on the other hand, have have more stringent delay constraints than loss constraints, leaving end-to-end delay and delay jitter as their major impediment to acceptable quality. Thus, interactive multimedia applications often prefer smaller queues in Internet routers, even if this means higher losses and lower throughput.

Unfortunately, current Internet routers do not provide a Quality of Service (QoS)[1] adapted to the current traffic mix. Most Active Queue Management (AQM) techniques are either heavy-weight by requiring significant architectural changes or focus on providing higher throughput at the router without much consideration for queuing delays.

RED [4], probably the best-known AQM mechanism, attempts to keep the average queue size at the router low while keeping throughput high. Adaptive RED (ARED) [2] adjusts the RED operating parameters to accommodate a wider range of traffic loads. However, both RED and ARED provide equal treatment to incoming traffic and tend to be tuned for high throughput without any consideration for the traversing applications' aggregate delay requirements.

By requiring per flow signaling and reservations via RSVP by all routers on the connection path, IntServ provides per flow QoS guarantees. However, the complexity of the RSVP signaling and the per flow soft state required at each router means that IntServ does not scale well with number of flows.

DiffServ approaches, such as Assured Forwarding (AF) and Expedited Forwarding (EF), provide differentiated service to traffic aggregates. However, they require complicated mechanisms to negotiate service level agreements. Additionally, DiffServ requires traffic monitors, markers, shapers, classifiers and droppers and a framework to enable these components to work together.

We propose an extension to ARED called called *RED-Worcester*[2] to improve overall QoS support at the router by satisfying the average performance requirements of incoming packets in terms of throughput and delay. While not a differential service, RED-Worcester provides a service that better matches the average requirements of all applications without adding much complexity and it requires no additional policing mechanisms, charging mechanisms or usage control.

2 RED-Worcester

In RED-Worcester, application end-hosts indicate their sensitivity to queuing delays as source hints that we refer to as *delay hints*. The delay hint is not an absolute bound on queuing delay, but rather indicates to Internet routers the relative importance of delay versus throughput for this flow. RED-Worcester does not guarantee queuing delays based on the delay hint, but uses the delay hint to calculate the average queuing requirements of incoming packets and adjust the target average queue accordingly. The idea being that higher average queue sizes typically have higher throughput while lower average queue sizes tend to have lower delays [1].

RED-Worcester directly extends Adaptive RED [2] (ARED). In ARED, the targeted average queue size is independent of traffic QoS requirements. We argue that the targeted average queue size should be based on the incoming traffic's requirements. Unlike ARED's fixed target queue size, RED-Worcester provides a moving target queue size based on an exponentially weighted moving average of the

[1] We use the term QoS to refer explicitly to *delay* and *throughput* provided by the network.

[2] Similar to various versions of TCP, which are named after cities in Nevada, we have named our RED active queue management technique after a city in Massachusetts.

delay hints of the incoming packets. Thus, when incoming traffic is mostly throughput-sensitive, RED-Worcester maintains a higher average queue to improve the overall throughput. On the other hand, when incoming traffic is mostly delay-sensitive, RED-Worcester lowers the average queue size to reduce the average queuing delays. In case of medium throughput and medium delay sensitive traffic, such as for HTTP traffic, RED-Worcester maintains an average queue size which gives medium throughput and delay performance.

3 Experiments

We have evaluated RED-Worcester in a variety of NS-2 simulations under traffic mixes that vary from mostly throughput-sensitive flows to mostly delay-sensitive flows [5]. Throughput-sensitive flows were represented by TCP flows carrying a delay hint of 80 ms. Delay-sensitive flows were represented by TFRC [3] flows carrying a delay hint of 32 ms.

Analyzing throughput and delay alone is not sufficient as application performance realistically involves tradeoffs between throughput and delay. We propose a new metric, *QoS*, that provides a quantitative performance metric designed to capture to some degree the nature of the throughput-delay tradeoff: $QoS = T^{\alpha}/D^{\beta}$, where T is throughput in bps, D is queuing delay in seconds at the router and $\alpha + \beta = 1$. While the choices of α and β depend upon the throughput and delay tolerances of applications themselves, the expectation is that the relativity of the delay hints from RED-Worcester enabled sources will vary accordingly. $\alpha = 1$ is used for throughput-sensitive flows (such as Email or file transfer), since these flows can tolerate delay but rely upon throughput. $\alpha = 0.5$ and $\beta = 0.5$ are used for delay-sensitive flows (such as interactive videoconferences), since these flows require low delay, but also moderate amounts of throughput. $0.5 < \alpha < 1$ and $0 < \beta < 0.5$ for medium throughput and medium delay-sensitive flows (such as HTTP for Web browsing). Note our new metric, QoS, is in fact a traffic-sensitive variant of the power performance metric.

Figure 1 shows the QoS performance of delay-sensitive flows in RED-Worcester normalized to the QoS of delay-sensitive flows in ARED. QoS for delay-sensitive flows improves in RED-Worcester as the fraction of delay-sensitive flows increases in the incoming traffic mix. RED-Worcester's moving target brings the average queue size down as the fraction of delay-sensitive flows increases, reducing the overall average queuing delays. On the contrary, in ARED, delay-sensitive flows suffer as ARED maintains the average queue size at a fixed target of 50 packets irrespective of the QoS constraints of the incoming traffic.

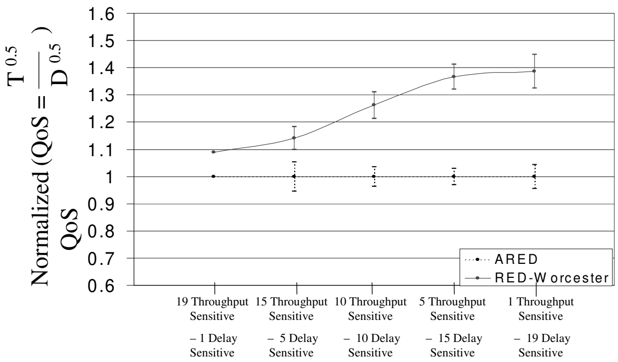

Figure 1. Normalized QoS

4 Summary

This paper briefly presents RED-Worcester, an active queue management technique designed to provide better overall Quality of Service (QoS). The service afforded by RED-Worcester does not require additional monitoring or charging of either flows that take advantage of the low delay service or flows that take advantage of the high throughput service, allowing it to integrate with the current best-effort Internet environment. Under the proposed service, RED-Worcester routers operate equally well under scenarios with only traditional traffic and operate better under scenarios with mixed or multimedia only traffic.

References

[1] V. Firoiu and M. Borden. A Study of Active Queue Management for Congestion Control. In *Proceedings of IEEE InfoCom*, Mar. 2000.

[2] S. Floyd, R. Gummadi, and S. Shenker. Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. Under submission, 2001. http://www.icir.org/floyd/papers/adaptiveRed.pdf.

[3] S. Floyd, M. Handley, J. Padhye, and J. Widmer. Equation-Based Congestion Control for Unicast Applications. In *Proceedings of ACM SIGCOMM Conference*, pages 45 – 58, 2000.

[4] S. Floyd and V. Jacobson. Random Early Detection Gateways for Congestion Avoidance. *IEEE/ACM Transactions on Networking*, Aug. 1993.

[5] V. Phirke, M. Claypool, and R. Kinicki. RED-Worcester – Traffic Sensitve Active Queue Management. Technical Report WPI-CS-TR-02-21, Worcester Polytechnic Institute, May 2002. ftp://ftp.cs.wpi.edu/pub/techreports/pdf/02-21.pdf.