Analysis of Receiver Window Control in Presence of a Fair Router

Vijay T. Raisinghani[*]
Tata Infotech Ltd.(ATG) and
KR School of Information Technology
IIT Bombay
rvijay@it.iitb.ac.in

Sridhar Iyer
KR School of Information Technology
IIT Bombay
sri@it.iitb.ac.in

ABSTRACT

Receiver Window Control is a mechanism used to change the TCP advertised window so as to manipulate the bandwidth sharing of multiple TCP sessions. The research on receiver window control so far [3],[5],[6] does not analyze behavior of receiver window in the presence of packet drops at a *fair* router. In this paper, we analyze receiver window control in the presence of a modified RED-DT (Random Early Detection with Dynamic Thresholds) [9] router. We show that receiver window control mechanism is *goodput preserving* and does not adversely impact competing traffic.

I. INTRODUCTION

TCP uses congestion and flow control mechanisms to avoid swamping the network or the receiver [2]. When the sender is not congestion window limited, the receiver can control the transmission rate of the sender by advertising a window, which reflects the buffer state at the receiver. This is the central idea in Receiver Window Control (RWC).

The current TCP implementations have fixed receive buffer sizes for all applications. Application level APIs are available, that allow an application to set its receiver buffer at the start of a connection [8]. However, once set it cannot be modified to reflect changes in application priorities.

Throughput for a TCP connection is decided by the receiver window setting and the corresponding bandwidth-delay product [7]. In case of multiple flows, each having a different bandwidth-delay product, each of the flows will have a different optimum receiver window(*awnd*). This property of *awnd*'s relation to the bandwidth-delay product can be exploited to *intentionally* make some of the TCP sessions get lower throughput, and thus dynamically control the application priorities. This assumes that total *actual* receiver buffer space is large enough to allow manipulation(increase or decrease) of the *awnd values* for the different sessions.

The intuitive benefits of using RWC can be seen from the following example: Consider a user running multiple downloads on a wireless device. The user increases the priority of a particular download. Through RWC, by decreasing the advertised window, the throughput can be decreased for the lower priority downloads. The bandwidth thus made available can be used by the higher priority download to increase its throughput. This control is dynamic and is invoked as and when the user changes application priorities.

Research about receiver window control so far [3], [5], [6] does not analyze receiver window control in the presence of packet drops at an intermediate router that ensures *fairness*. In this paper we discuss RWC in in the presence of a RED-DT (Random Early Detection with Dynamic Thresholds) router [9]. We also show that RWC is *goodput preserving* and that it does not adversely affect competing traffic.

We provide a brief introduction to RED-DT. A RED-DT router implements an enhancement to Random Early Detection (RED) [1]. RED-DT ensures fairness even in the presence of unresponsive flows. In contrast to RED which has a single queue, fixed thresholds and single drop probability for all flows, RED-DT manipulates *queue thresholds* and *drop probabilities* for *each* flow. For each flow i, it maintains average queue size $q_{i_{ave}}$, queue thresholds max_{th} and min_{th}, drop probabilities p_i (instantaneous) and $p_{i_{max}}$ (maximum) and a weight $\alpha < 1$. α is larger for flows identified as greedy. On packet arrival, $max_{th} = (1 - \alpha p_{i_{max}}) \times available\ buffer$. Instantaneous drop probability p_i calculation is similar to RED, except that $q_{i_{ave}}$ is used. Flows having a $q_{i_{ave}}$ higher than the aggregate average queue size are identified as greedy. $p_{i_{max}}$ is gradually increased for greedy flows, and decreased for responsive flows. This ensures fair bandwidth sharing among the flows.

The organization of our paper is as follows: in section II we present the analysis of receiver window control and in section III we summarize the paper and provide an overview of the related work.

II. RECEIVER WINDOW CONTROL ANALYSIS

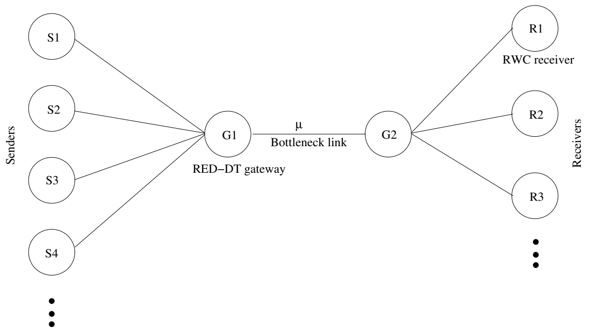

Fig. 1. Receiver Window Control with RED-DT router

Our analysis is based on the model shown in figure 1. We consider a set of senders Si sending data to receivers Ri over a bottleneck link $G1 - G2$ with constant capacity μ through a RED-DT router ($G1$) [9]. One of the receivers, $R1$, implements receiver window control.

[*]This author is a Ph.D student at IIT Bombay, sponsored by Tata Infotech Ltd.

Let, F be the set of flows that terminate on R1. Fair share for a flow is μ/n, where n is the number of active flows over $G1-G2$. We define fair share for F as $\mu_F = \frac{\mu}{n}|F|$.

For our analysis we use a modified RED-DT router which
• identifies flows belonging to F by destination address of the packets
• manipulates its parameters ($max_{th}, p_{i_{max}}$ and α) such that μ_F is constant when n is constant. This means that if one flow in F decreases its throughput other flows in F are allowed to increase their throughput, to ensure that μ_F remains constant. This requires some additional book-keeping to be done by the router. We do not go into the details of modifying RED-DT.

Since the flows are packetized, $\mu'_F = \mu_F \pm \epsilon$, where μ'_F is the fair share after manipulation of RED-DT parameters and ϵ is the rounding error. This leads to the following observation:

Observation 1: $\mu'_F \approx \mu_F$

Assumptions

• Any packet loss is due to the RED-DT router. No packet losses occur on any of the links.
• All flows terminating on $R1$ are long-term bulk transfer TCP flows.
• The packet size, RTT and timeout of all the flows is the same.

Background

If every alternate packet is acknowledged the long-term steady state goodput (in packets) of a flow i is given by the following [4] :
If $W(p_i) < A_i$ then

$$T(p_i) = \frac{\frac{1-p_i}{p_i} + \frac{W(p_i)}{2} + Q(p_i, W(p_i))}{RTT(W(p_i)+1) + \frac{Q(p_i,W(p_i))G(p_i)T_o}{1-p_i}} \quad (1)$$

otherwise

$$T'(p_i) = \frac{\frac{1-p_i}{p_i} + \frac{A_i}{2} + Q(p_i, A_i)}{RTT(\frac{A_i}{4} + \frac{1-p_i}{p_i A_i} + 2) + \frac{Q(p_i,A_i)G(p_i)T_o}{1-p_i}} \quad (2)$$

where, p_i is loss probability for the flow, RTT is round trip time, T_o is timeout, A_i is maximum advertised window, $W(p_i)$ is expected congestion window size for flow i, $Q(p_i, w)$ is the probability that a loss in a window of size w is due to a timeout, and $\frac{G(p_i)T_o}{1-p_i}$ is the average duration of a timeout sequence. $W(p_i), Q(p_i, w)$ and $G(p_i)$ are as shown in equation (3),(4),(5)

$$W(p_i) = \frac{2}{3} + \sqrt{\frac{4(1-p_i)}{3p_i} + \frac{4}{9}} \quad (3)$$

$$Q(p_i, w) = min(1, \frac{(1-(1-p_i)^3)(1+(1-p_i)^3(1-(1-p_i)^{w-3}))}{1-(1-p_i)^w}) \quad (4)$$

$$G(p_i) = 1 + p_i + 2p_i^2 + 4p_i^3 + 8p_i^4 + 16p_i^5 + 32p_i^6 \quad (5)$$

Estimating receiver window value:
Receiver window control throttles one or more senders to increase the bandwidth share of other flows. The goodput of a sender will be determined by equation (1) or (2) depending on the size of the receiver window relative to the sender's congestion window. To effect receiver window control on a sender, the receiver needs to estimate the sender's congestion window and then calculate the advertised window depending on the priority of the session. Since, from equations (1 - 5) it can be seen that $T'(p_i) < T(p_i)$. The advertised window needs to be set such that equation (2) determines the sender's goodput. Thus, for the flow i that needs to be throttled, the receiver sets A_i such that $A_i < W(p_i)$.

The receiver can estimate p_i, and thus $W(p_i)$, from the missing packets.

Using the above background we now analyze receiver window control (RWC).

Theorem 1: **RWC results in changed bandwidth share of flows in F**

Proof: At steady state, let μ_F and μ'_F be the aggregate goodput at the receiver before and after RWC, respectively. In a time period, when no flows are added or removed, the RED-DT router attempts to maintain μ_F at a constant value.

For a flow i let,
• τ_i represent the throughput. $\tau_i = T(p_i)$ (equations (1), (2)).
• ω_i represent the congestion window. $\omega_i = W_i(p_i)$ (equation (3)).

Let F contain two flows. Thus,

$$\tau_1 + \tau_2 = \mu_F \quad (6)$$

Initially let $A_1 > \omega_1$ and $A_2 > \omega_2$. RWC is initiated by setting $A_1 < \omega_1$ and $A_2 > \omega_2$. A_1 is constant.

After RWC, let
• p'_1 and p'_2 be the loss probabilities at the RED-DT
• τ_{1_b} and τ_{2_b} be the goodput, *before* loss probability change at RED-DT router
• τ'_1 and τ'_2 be the goodput, *after* loss probability change at RED-DT router
• ω'_1 and ω'_2 be the expected congestion windows, *after* loss probability change at RED-DT router.

When RWC is initiated, since $A_1 < \omega_1$, from equation (1 - 2) $\tau_{1_b} < \tau_1$. $\tau_{2_b} = \tau_2$, since $A_2 > \omega_2$ and loss probabilities have not been changed at the RED-DT router. Hence on RWC initiation $\mu'_F < \mu_F$.

To ensure constant fair share μ_F, the RED-DT router decreases p_1 to p'_1 and p_2 to p'_2 in an effort to increase μ'_F to μ_F. Thus eventually,

$$\tau'_1 + \tau'_2 = \mu_F \quad (7)$$

Since, $p'_1 < p_1$ and A_1 is constant, from equation (2) $\tau'_1 = \tau_{1_b}$ i.e. $\tau_1 - \tau'_1 = \delta$ $(constant)$.

Since, $A_2 > \omega'_2$, from equation (1) $\tau'_2 > \tau_2$.

From equation (6) and (7), $\tau'_2 = \tau_2 + \tau_1 - \tau'_1$ or $\tau'_2 = \tau_2 + \delta$. □

Example: Let, $p_1 = p_2 = 0.001$. Then from equations (1 - 5), $\tau_1 = \tau_2 = 27$; $\omega_1 = \omega_2 = 37$. Thus from equation (6), $\mu_F = 27 + 27 = 54$. If $A_1 = 18$, $\tau_{1_b} = 16$. To ensure that μ_F remains constant, the RED-DT router would decrease p_1 and p_2. At $p'_1 = p'_2 \approx 0.00055$ $\tau'_2 = 36$, $\tau'_1 = 16$. $\mu'_F = 52 (\approx \mu_F)$, which is in-line with observation 1.

Theorem 2: **The aggregate goodput of flows over the bottleneck link external to RWC is unaffected by RWC.**

Proof: At steady state, let μ_E and μ'_E be the goodput of the flows before and after RWC respectively. $\mu_E = \mu - \mu_F$ and $\mu'_E = \mu - \mu'_F$. From observation (1) $\mu'_F \approx \mu_F$. Thus $\mu'_E \approx \mu_E$. $\square$

From the above, we can see that receiver window control is *goodput preserving* and *friendly* to other flows.

III. SUMMARY

Receiver window control helps a receiver to partition its bandwidth share appropriately among its TCP sessions. However, receiver window control is affected by packet drops at routers.

We analyzed the behavior of receiver window control in presence of packet drops at a RED-DT router. We showed that RWC is *goodput preserving* and *freidnly* to competing traffic.

Receiver window control has been proposed and studied by [3], [5], [6] and other references therein. However, they do not analyze the behavior in presence of a *fair* router like RED-DT. [6] discuss how receiver window control can be used to decrease congestion at the router and thus prioritize flows. [3] and [5] focus on user feedback for determining receive buffers and thus manipulate bandwidth sharing. [3] also use delayed ACKs in conjunction with RWC. [3] and [6] also present an implementation of receiver window control.

Other issues that can effect receiver window control, such as the effect of packet drops on the link, require further study.

REFERENCES

[1] S. Floyd and V. Jacobson. Random Early Detection Gateways for Congestion Avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, 1993.

[2] M. Allman and V. Paxson and W. Stevens. RFC2581: TCP Congestion Control, Apr. 1999.

[3] P. Mehra, A. Zakhor, and C. D. Vleeschouwer. Receiver-Driven Bandwidth Sharing for TCP. In *INFOCOM*, 2003.

[4] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose. Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation. *IEEE/ACM Transactions on Networking*, 8(2):133–145, 2000.

[5] V. T. Raisinghani, A. K. Singh, and S. Iyer. Improving TCP Performance over Mobile Wireless Environments using Cross Layer Feedback. In *IEEE ICPWC*, New Delhi, India, Dec. 2002.

[6] N. Spring, M. Chesire, M. Berryman, V.Sahasranaman, T. Anderson, and B. Bershad. Receiver Based Management of Low Bandwidth Access Links. In *INFOCOM*, 2000.

[7] W. R. Stevens. *TCP/IP Illustrated, Volume I, The Protocols.* AWL, 1994.

[8] Von Welch. A User's Guide to TCP Windows. `http://archive.ncsa.uiuc.edu/People/vwelch/net_perf/tcp_windows.html`, 1996.

[9] V. Vukadinovi and L. Trajkovi. RED with Dynamic Thresholds for Improved Fairness. In *Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 371–372. ACM Press, 2004.