

Analyzing the Receiver Window Modification Scheme of TCP Queues

Visvasuresh Victor Govindaswamy
University of Texas at Arlington
Texas, USA
victor@uta.edu

Gergely Záruba
University of Texas at Arlington
Texas, USA
zaruba@uta.edu

G. Balasekaran
University of Pittsburgh
Pennsylvania, USA
bala@hgen.pitt.edu

Abstract – Explicit Congestion Notification (ECN) and Active Queue Management (AQM) Schemes such as Random Early Detection (RED), Adaptive Random Early Detection (ARED) and BLUE queues have been proposed for TCP/IP networks to compensate network congestion. However, using ECN requires that ECN be supported by both TCP senders and receivers. This paper presents a novel AQM modification called Receiver-Window Modification (RWM). RWM can be used together with RED, ARED and BLUE queues, to provide congestion avoidance in packet switched networks at ingress and gateway routers. RWM does not require modification to all end system TCP/IP stacks but can be solely implemented in routers. Our RWM scheme helps in reducing the average queue sizes of RED, ARED, BLUE and even ARED-ECN, BLUE-ECN and RED-ECN queues. By reducing the average queue sizes, RWM queues reduce the queuing delay resulting in significant improvements in one-way end-to-end packet delays and dropped packets. It is also shown that the performance of RED-ECN, ARED-ECN and BLUE-ECN queues is heavily dependent on the queues of the downstream routers. RWM modified queues in ingress or gateway routers are not influenced by the number and state of the downstream router as they will piggyback congestion information to the source in the next available acknowledgement packet. We carry out extensive ns2 simulations to show our results and to support our claims.

Keywords–TCP; Active Queue Management (AQM); Buffer Management; Random Early Detection (RED); Congestion Control; Congestion Avoidance.

I. INTRODUCTION

TCP (Transmission Control Protocol) is the major connection oriented *transport layer protocol* used in today's Internet[1], where in the presence of bursty Internet traffic, routers along the path may buffer packets in their queues to absorb/reduce the burstiness. Queuing theory implies that the bigger the queue's capacity is, the more a queue can deal with the burstiness without dropping a packet. Unfortunately, TCP's congestion control can introduce a high variance (resulting in a high delay jitter) in the current buffer sizes. If the maximum buffer size of routers is increased, the TCP senders will be notified about congestion much later. To reduce these problems, Active Queue Management schemes such as RED[2], ARED[3] and BLUE[4] were introduced.

This paper deals with queue-based AQM, presenting a novel approach called Receiver Window Modification (RWM) for congestion avoidance in TCP. RWM may work with RED, ARED and BLUE algorithms together relaxing their need for a modified TCP layer at servers and clients. Our approach is specifically designed for ingress and gateway routers (see item 3. of Section IV with regards to core routers). The main idea of the approach is to restrict the TCP transmission window with the flow control window instead of the congestion control window, thus controlling the transmission window with a finer granularity. Results from simulations show that the RWM modified queues improve on the average queue size, one-way packet delay, delay jitter, number of packet drops and throughput as compared to RED, ARED, BLUE, RED-ECN, ARED-ECN and BLUE-ECN queues, especially in paths that have non-ECN compliant routers. Moreover, RWM does not require modification to TCP implementations at servers or clients, i.e., no "RWM-compliance" is needed.

The rest of the paper is organized as follows: Section II introduces the RWM scheme; simulation details, results as well as analysis are covered in Section III. The paper is concluded and future research directions are outlined in Section IV.

II. RED-RWM SCHEME

AQM solutions, such as RED, use packet drops at router queues to manipulate the TCP sender into decreasing its transmission rate. With ECN-compliant RED queues, packets are not unnecessarily dropped but marked as if they were dropped. However, as a downside: i) ECN messages may get lost or dropped by downstream routers; and ii) TCP implementations at both the source and the destination have to be ECN-compliant (which presents a significant problem in today's implementations). Currently there is no practical benefit in setting ECN bits in Internet packets; as this requires ECN capable routers (at least at bottleneck points), and servers and clients throughout a network. In [5], experiments were conducted using TBIT (the TCP Behavior Inference Tool) showing that in September 2000 only 21 out of 26447 (0.07%) sites positively responded with an

appropriate SYN/ACK. In March 2002, only 7 out of 12364 sites (0.05%) responded positively. These tests included sites running nearly all types of operating systems.

We are proposing to use flow control feedback to reduce congestion at routers. In this paper, we will deal with congestion occurring at the ingress and gateway routers – two major congestion areas within the network (see item 3. of Section IV with regards to core routers). Instead of setting ECN related bit in the chosen packets of RED, ARED and BLUE queues, we propose the use of receiver window modification (RWM). RWM sets the receiver window field to one maximum segment size (MSS) in the ACK packets¹ that are going towards the sender from the receiver, instead of dropping or marking the packets at the queue. (The only exception that it will not modify the receiver window field is when the field has a value of 0. This occurs when a TCP application wants to tell its peer not to send any more data). We denote RWM modification to AQM schemes by affixing “RWM” after the name of the AQM, thus we will be talking about RED-RWM, ARED-RWM and BLUE-RWM respectively. In the case of RED-RWM and ARED-RWM, the field is set to 1 only if the average queue length is between Th_{min} and Th_{maz} , where Th_{min} and Th_{maz} , are the minimum and maximum threshold values respectively. However, if the average queue length is greater than Th_{maz} , the packet is dropped. These thresholds are the same as in RED and ARED. In the case of BLUE-RWM, it uses the probability, P_m , instead of marking (by using ECN) or dropping queued packets (as in BLUE), to modify the ACK packets. P_m is increased whenever ACK packets are modified or packets are dropped from the queue and decreased when the link is underutilized. The state in RWM is kept to a minimum since an ACK packet is modified per moment at which otherwise a packet will be dropped, i.e. the first after otherwise a packet will be dropped. Upon receiving the modified ACK packet, the sender will transmit the minimum of the congestion and the advertised received window sizes. (Whenever the TCP header of an ACK packet is modified, the checksum in the TCP header needs to be adjusted for error control.) A model of this scheme and its analysis has been presented in [5].

The advantages of RWM queues are that they work with the current implementation of TCP in end systems and do not require changes to the existing network schemes except at the ingress and gateway routers. Hence, they overcome the disadvantage of the ECN queues since in ECN, TCP/IP stacks at both ends require modifications to be “ECN-compliant”. Since, the feedback loop extends from the RWM queue on a router to the sender; the response of the RWM scheme is determined by the delay between the router and the sender rather than the round-trip times of the connection, which is true in the case for the queues using the ECN mechanism. The delayed response of the sender to an ECN is further compounded if the congestion worsens at

the other routers lying between the router and the receiver and between the receiver and the sender. The response time will also be heavily dependent on the types of queue implementations at these routers. The queues using the RWM mechanism enjoy the advantage of faster response times to the impending congestion since the notification of congestion arrives at the source quicker when compared to those using ECN mechanism.

III. SIMULATION RESULTS

We have conducted an extensive simulation experiment to evaluate and compare AQM algorithms without ECN, with ECN and with our proposed RWM scheme. Our simulations were based on an extended and corrected (the receiver window – congestion window interaction in NS2 does not follow a common linux-type TCP implementation) NS2 [7] simulator. The subsequent subsections explain the details and discuss the results of the different simulation scenarios. We compare AQM schemes on the simple network topology of 4 sources connected to a sink via 4 routers as shown in Fig. 1; the nearest router to the sources employs the AQM to be investigated while the other 3 routers contain Drop Tail queues which are kept large enough not to discard any packets. We had used TCP Reno as most modern TCP’s are “Reno” based.

A. Scenario 1 – RED AQM

We first study the performance of TCP with RED, RED-ECN and RED-RWM queues. During the interval (0, 150) seconds, four File Transfer Protocol (FTP) connections are started at time $t = 0s$ with the size of the queue set at 35 packets. The RED parameters were set as follows:

- Queue weight given to current queue size sample = 0.002
- $Th_{min} = 5$ and $Th_{maz} = 15$
- Max probability of dropping a packet = $1/linterm$, where $linterm = 3$

Simulation results were obtained for RED, RED-ECN and RED-RWM AQMs employed in the first router. Fig. 2 depicts the average queue sizes versus the simulation time of the investigated AQMs. RED-RWM reduces the average queue size by about 25% compared to RED and RED-ECN. In the case of the RED-ECN queues, the larger queue sizes

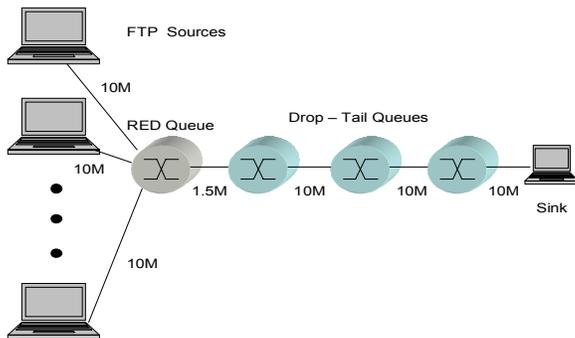


Figure 1. Simulation network topology.

¹ Some researchers claim that routers in general should not look at layer-4 headers as their function is limited to layer-3. Yet, enabling them to do so can provide significant benefits (see e.g., NAT).

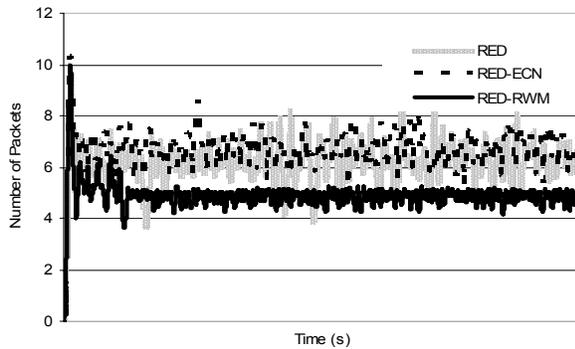


Figure 2. The average queue sizes for RED, RED-ECN and RED-RWM.

are partly due to marked packets suffering congestion at the downstream routers, increasing the delay along their way to the destination and then being echoed back to the source via ACK packets. On the other hand, the feedback loop for the RED-RWM is much shorter and hence, its average queue size is closer to the minimum threshold of average queue size (Th_{min}) of 5 packets. When the buffering delay is increased, the corresponding round-trip times increase causing the aggregate TCP behavior to be less aggressive. Likewise, when the buffering delay is decreased, the corresponding round-trip times decrease causing the aggregate TCP behavior to be more aggressive.

Keeping the same topology and RED parameters but varying the number of sources, RED, RED-ECN and RED-RWM, we investigate their performance by collecting essential data and analyzing them in terms of i) delay jitter (the variation in the time taken for packets to be transmitted from the source to the destination in a network.); ii) average packet delay, iii) number of packets dropped; and iv) throughput. Figs. 3, 4, 5 and 6 show the respective results obtained for 1000 seconds of simulation time. The results confirm that RED-RWM outperforms both RED and RED-ECN in all of the above metrics. The average packet delay of RED-RWM is 15-20% less than that of RED-ECN and RED. Moreover, RWM significantly and consistently reduces the variation of the delay by about 35%. Our scheme also outperforms its counterparts in terms of overall throughput and packet drops.

Smaller queue sizes imply smaller queuing delay and as a result, there is an improvement in the one-way packet delay with RED-RWM as compared to those obtained with RED-ECN and RED queues. The duration of the reception of three duplicate acknowledgements (ACKs), due to early-dropped packets, as in the case of RED, delayed the response time of the TCP congestion scheme causing its larger average queue size. This observation was also noticed when we had used timeouts in our experiments. Moreover, delay jitter for RED-RWM is less since its steady-state queue size is more stable than those of RED and RED-ECN. Since RED-RWM does not drop packets when its average queue size is between the two thresholds, Th_{min} and Th_{max} , it obviously drops fewer packets than RED. As for RED-

ECN, the delayed arrivals of marked packets delay the action of TCP congestion scheme in controlling the congestion, thereby increasing the number of packet drops at the router. Similar results were obtained for RWM schemes in Scenarios 2 and 3 when compared to ARED, ARED-ECN, BLUE and BLUE-ECN.

B. Scenario 2 –ARED AQM

Here, we investigate the performance of ARED, ARED-ECN and ARED-RWM queues using the network topology and RED parameters of Scenario 1. The average queue sizes for ARED, ARED-ECN and ARED-RWM are compared with each. Generally, it can be observed that the average queue size for ARED-RWM is smaller than those of ARED and ARED-ECN (Fig. 7 - The average queue size curves of ARED and ARED-ECN are comparable and semi-overlapping). This is due to the faster response time to the impending congestion since the notification of congestion for ARED-RWM queue arrives at the source faster when compared to the ECN (in the case of ARED-ECN) and the delay added on to the round-trip times by the duration of the reception of three duplicate acknowledgements or by those caused by the timeout mechanisms due to each packet drop (in the case of ARED). Note, that for the ARED-ECN queue the notification is delayed by all of the queuing delays of the downstream routers. Hence, the average queue sizes for ARED-RWM are about 50% smaller compared to those of ARED-ECN and ARED.

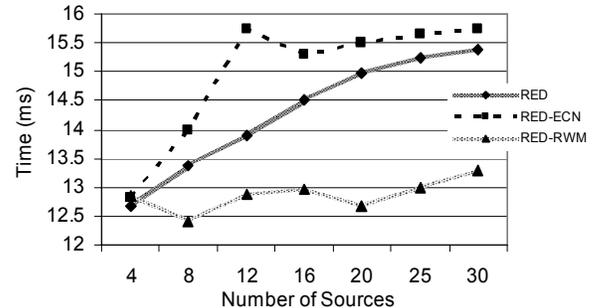


Figure 3. Average packet delay for RED, RED-ECN and RED-RWM.

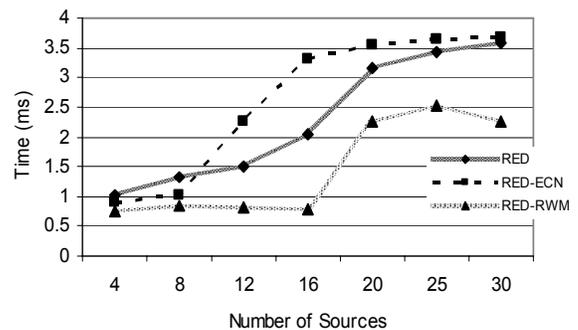


Figure 4. Delay Jitter for RED, RED-ECN and RED-RWM.

Keeping the same topology and RED parameters but varying the number of sources (increasing the duration of the experiments to 1000s), ARED, ARED-ECN and ARED-RWM queues were compared on their one-way packet delay, delay jitter, number of packet drops and throughput. Figs. 8, 9, 10 and 11 show the results that confirm that ARED-RWM outperforms ARED and ARED-ECN in all of these categories. The analyses for these are the same as those in the Scenario 1.

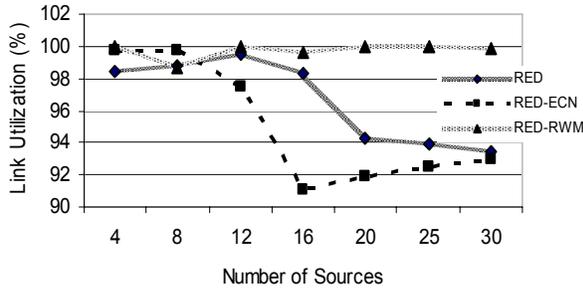


Figure 5. Throughput for RED, RED-ECN and RED-RWM.

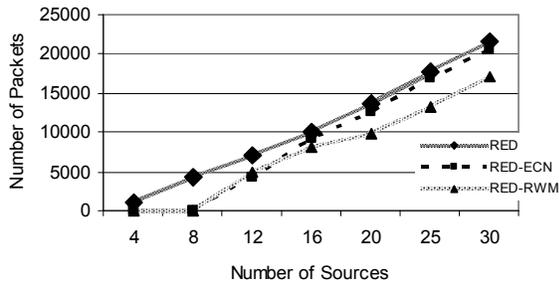


Figure 6. Total packet drops for RED, RED-ECN and RED-RWM.

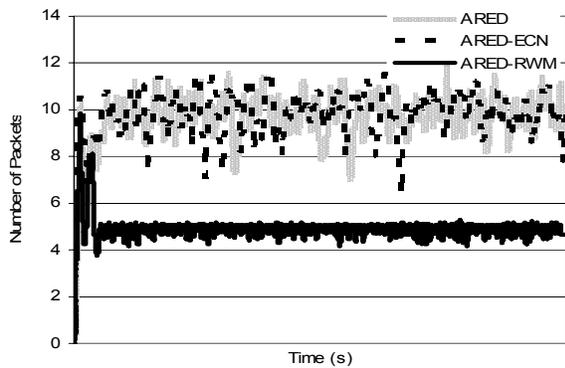


Figure 7. The average queue sizes for ARED, ARED-ECN and ARED-RWM.

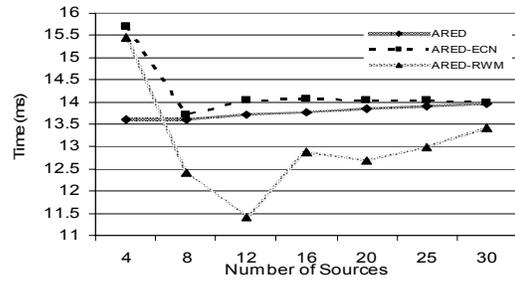


Figure 8. Average packet delay for ARED, ARED-ECN and ARED-RWM.

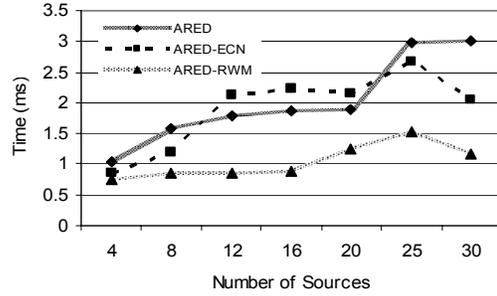


Figure 9. Delay jitter for ARED, ARED-ECN and ARED-RWM.

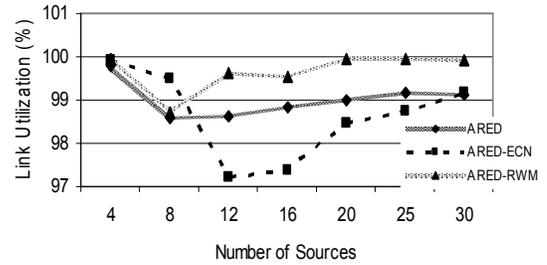


Figure 10. Throughput for ARED, ARED-ECN and ARED-RWM.

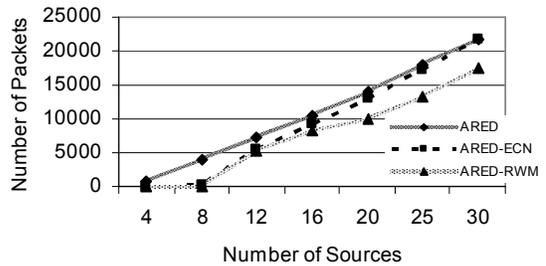


Figure 11. Total packet drops for ARED, ARED-ECN and ARED-RWM.

C. Scenario 3 –BLUE AQM

Here, we investigate the performance of BLUE, BLUE-ECN and BLUE-RWM queues using the network topology of scenario 1. We use the following BLUE parameters [4]:

$Freeze_time = 10ms$, $\delta_1 = 0.02$, $\delta_2 = 0.002$ and threshold $L = 15$ packets. The average queue sizes for BLUE, BLUE-ECN and BLUE-RWM are shown in Fig. 12. The average queue size for BLUE-RWM is slightly better than those of BLUE and BLUE-ECN.

Keeping the same topology and BLUE parameters but varying the number of sources, BLUE, BLUE-ECN and BLUE-RWM queues are evaluated in terms of their one-way packet delay (Fig. 13), delay jitter (Fig. 14), throughput (Fig. 15) and number of packet drops (Fig. 16). In terms of the packet delay, delay jitter, number of packet drops and throughput, BLUE-RWM significantly outperforms BLUE and BLUE-ECN. The analyses for these are the same as those in the Scenario 1.

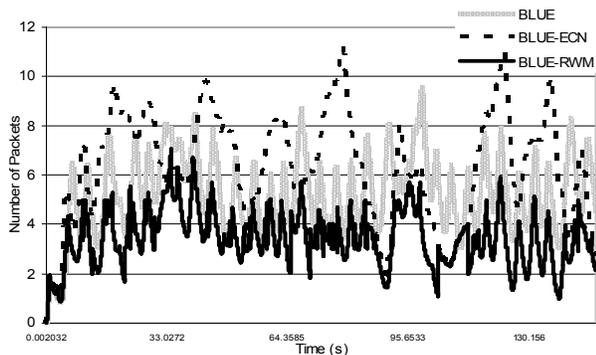


Figure 12. The average queue sizes for BLUE, BLUE-ECN and BLUE-RWM.

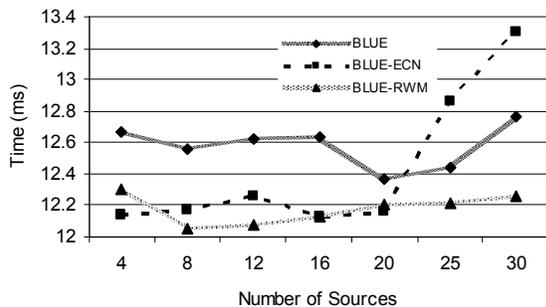


Figure 13. Average packet delay for BLUE, BLUE-ECN and BLUE-RWM.

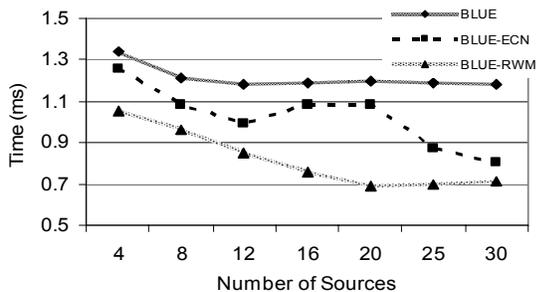


Figure 14. Delay jitter for BLUE, BLUE-ECN and BLUE-RWM.

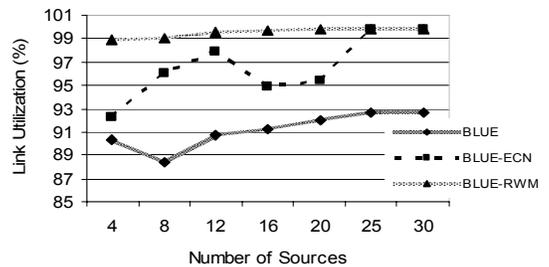


Figure 15. Throughput for BLUE, BLUE-ECN and BLUE-RWM.

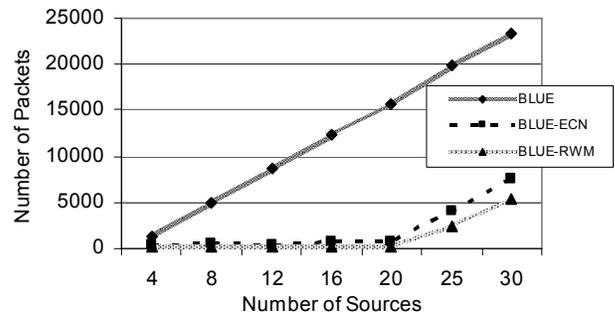


Figure 16. Total packet drops for BLUE, BLUE-ECN and BLUE-RWM.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a modification to existing adaptive queue management protocols called Receiver Window Modification (RWM). RWM does not require modification to all end system TCP/IP stacks but can be solely implemented in routers (especially for ingress and gateway routers). This means that RWM does not require both the sources and receivers to be “RWM-compliant” as was the case for ECN-compliant queues. Our RWM scheme helps to reduce the average queue sizes of the ARED, BLUE, RED, ARED-ECN, BLUE-ECN and RED-ECN queues. By reducing the average queue sizes, RWM queues reduce the queuing delay resulting in significant improvements in one-way end-to-end packet delays and dropped packets. It is also shown that the performance of RED-ECN, ARED-ECN and BLUE-ECN queues is heavily dependent on the queues of the downstream routers. RWM queues are not influenced by the number and state of the downstream router as they will piggyback congestion information to the source in the next available acknowledgement packet.

We are currently working on research to

1. Implement RWM in Linux based packet routers and by using various flavors of TCP with multi-bottleneck topologies.
2. Control congestion caused by non-TCP flows in order to complement the gains obtained by using the RWM scheme for TCP flows.

3. Control congestion with an extended-RWM (ERWM) scheme at the edge routers to ensure that at the borders of the network that each flow's packets do not enter the network faster than they are able to leave it. The aim here is to push any complexity in controlling congestion toward the edges of the network by not modifying the core routers and the end systems.

REFERENCES

- [1] S. Floyd, "Measurement Studies of End-to-End Congestion Control in the Internet," <http://www.icir.org/floyd/ccmeasure.html>.
- [2] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking, pp. 397-413, vol. 1, no. 4, August, 1993.
- [3] S. Floyd, R. Gummadi, and S. Shenker. "Adaptive RED: an algorithm for increasing the robustness of RED's Active Queue Management", <http://www.icir.org/floyd>
- [4] W. Feng, D. Kandlur, D. Saha, and K. Shin, "The BLUE active queue management algorithms", IEEE/ACM Transactions on Networking (TON) Volume 10, Issue 4 (August 2002) Pages: 513 – 528, 2002.
- [5] TBIT, the TCP Behavior Inference Tool. <http://www.icir.org/tbit/ecn-tbit.html>.
- [6] V.V. Govindaswamy, G. Zaruba, G. Balasekaran, "Receiver-Window Modified Random Early Detection (RED-RWM) Active Queue Management Scheme: Modeling and Analysis", Accepted for Publication-2006 IEEE International Conference on Communications (ICC 2006), Istanbul, Turkey, 11-15 June 2006.
- [7] NS simulator, <http://www.isi.edu/nsnam/ns/>