Analytic Understanding of RED Gateways with Multiple Competing TCP Flows

Alhussein Abouzeid, Sumit Roy
Department of Electrical Engineering, University of Washington
Box 352500, Seattle, WA 98195-2500, USA
e-mail: {hussein,roy}@ee.washington.edu

Abstract— **An analytical framework for multiple TCP flows sharing a bottleneck link under the Random Early Detection (RED) regime is developed. Closed form expressions for the steady state throughput and average queueing delay are derived and verified by simulations; these show that RED significantly improves the inherent TCP bias against links with higher round-trip delays as compared to Tail Drop, contrary to prevailing belief. Further, we derive closed form bounds on the minimum average queuing delay achievable through a RED gateway with no deterministic packet drop.**

I. Introduction

This work aims to make a contribution towards modeling of multiple TCP flows sharing a (bottleneck) RED gateway, consistent with recent interest in developing analytical models for the dynamic behavior of TCP. Such efforts contribute significantly to verifying insights obtained from many simulations in a number of ways as well as overcoming the difficulty of developing simulation code for practical scenarios with large number of design/system parameters. Analytic results also provide insight as to how non-responsive flows may be made 'TCP-friendly', an objective of recent work.

Active queue management has been recently proposed as a means to alleviate some congestion control problems as well as provide a notion of quality of service. In traditional (passive) Tail Drop gateways, packet drop takes place only when packets arrive to a full buffer. When multiple TCP sessions sharing a common gateway arrive at a full buffer, the impact is (typically) simultaneous packet loss and reduction of transmission rate in *all* flows, resulting in an oscillatory behavior termed *global synchronization* reported experimentally by [1],[2]. This leads to link capacity under-utilization and exacerbates the inherent bias of TCP congestion control algorithm against higher delay flows. Accordingly, packet drop policies alternate to Tail Drop have been sought - a promising class is based on randomized packet drop [3], [4] (or marking [5], [6]) that avoid such global synchronization resulting from deterministic drop.

In Random Early Detection (RED) [4], packets are dropped when the (exponentially weighted) time-averaged queue size exceeds min_{th}, with a probability that increases linearly until it reaches max_p at average queue size max_{th}; subsequently all packets are dropped w.p. 1. While 'optimum' values of RED parameters as well as enhancements to the preliminary RED algorithm [4] are still subject to ongoing research (e.g. [7], [8]), the key objective of RED can be stated as follows: *packet drop (i.e. congestion notification) must be generated at a rate sufficient to prevent packet loss due to buffer overflow (otherwise, RED will act as a Tail Drop) while sustaining high link utilization.*

This work was supported in part by AFOSR Grant F49620-1-0472

In view of the Internet Engineering Task Force's (IETF) informational RFC [9] urging deployment of RED in the Internet, and the fact that TCP supports 90% of Internet packet transport [10], understanding of the interaction between multiple TCP flows and RED is a key requirement for successful end-to-end congestion control in the Internet.

The main objectives of this paper are to present a simple analytical model for RED gateways with multiple competing TCP flows and derive *closed form* expressions for the *steady state* average TCP window size, rate and average queue size as a function of gateway parameters and the packet drop (or marking) rate of the gateway.

While a number of analytic models for TCP congestion control over Tail Drop gateways exist that vary in their scope of application (e.g. number of flows) and accuracy (e.g. [11], [12], [13], [14], [15], [16]), similar relevant efforts for RED gateways [17], [18], [19], [20] are sparse. The work of [18], [19] do not incorporate essential window dynamics of TCP, i.e., the decrease of the window size in response to congestion, and the effect of implicit (packet loss detection) or explicit (ECN) notification of congestion. We underscore the significance of our new *closed form* results (verified with *ns* ([21]) simulations) that highlights the inaccuracy of the prevailing claim, heuristically argued in various occasions in the literature (e.g. [4]) that randomization of packet drop at a congested gateway has little effect on the bias in TCP against high delay links. While the above is valid for Tail Drop, *random drop policies in congested gateways significantly decrease the TCP bias against links with high delay.* The closed form expressions also allow us to derive a lower bound on the minimum achievable queuing delay through a RED gateway.

The paper is organized as follows. In Section II, we present a model for a network with multiple TCP flows sharing a bottleneck link and outline our modeling approach and the approximations used. In Section III, we derive simple closed-form analytic expressions for various parameters of interest listed above.

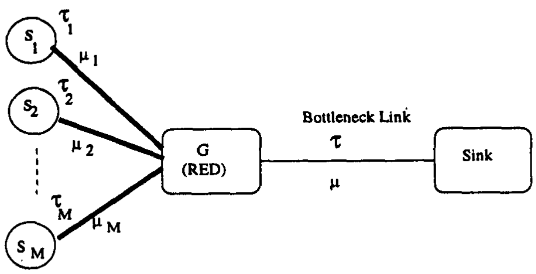

Fig. 1. Schematic of multiple TCP connections sharing a bottleneck link.

These results are validated in Section IV against *ns* simulations. Section V concludes the paper, outlining future extensions.

II. A Model for Multiple TCP Flows

A. System Model

Figure 1 depicts the system model considered. M TCP flows share a common bottleneck link with capacity μ packets/sec and (one-way) propagation delay τ. Forward TCP traffic is a one way flow between a source S_j and the sink, with reverse traffic consisting only of packet acknowledgments (ACKs) for each successfully received packet at the sink. We consider the long run (steady-state) behavior and assume that the sources always have packets to transmit (e.g. FTP of large files). The j-th flow encounters a one-way propagation delay τ_j, and is connected via a link with capacity μ_j packets/sec. to the bottleneck gateway. The total round-trip time for propagation and transmission (but excluding queuing) for the j-th flow is thus

$$T_j = \frac{1}{\mu_j} + 2(\tau_j + \tau) + \frac{1}{\mu} \tag{1}$$

We consider the Reno version of TCP and assume the reader is familiar with the key aspects of the window adaptation mechanism: the two modes of window increase (slow start and congestion avoidance) and the two modes of packet loss detection (reception of multiple ACKs with the same next expected packet number or timer expiry). We assume the receiver sends an acknowledgment for each packet (i.e. no delayed ACKs).

B. Modeling Approach

The primary challenge facing any analytical approach is that of capturing all the complex details of TCP protocol and active queue management policies as RED, mainly due to the extent of memory retained in these systems. Fortunately, in some cases, a mean value analysis approach can capture the essential dependencies among the various parameters in the modeled system. While such (mean value) analysis does not allow for prediction of higher order statistical information (e.g. variances), this disadvantage is offset by the success of this method in providing simple analytical models to complex systems (e.g. [13], [16], [22], [23]). In our work, the mean value analysis is combined with the 'independence' assumption for dependent variables, and also invokes a fixed point approximation whereby $E[f(X)] \approx f(E[X])$.

We state and discuss the following approximations:

A1: Packet losses are always detected with triple duplicate AKCs.

A2: The time delay between packet loss and its notification at the source (on the order of one round-trip time) is neglected.

A3: The bottleneck link is fully utilized (except possibly at some few isolated instants).

A4: The only source of packet loss is the RED packet drop policy.

A1 allows us to neglect time-outs and slow-start effects and results in TCP sessions operating always in the congestion avoidance mode. The analysis of the cases with frequent time-outs is postponed to future extensions. While **A2** is invoked primarily for presentation simplicity, we will show later that it has negligible effect on the model accuracy. **A3** is reasonable if the TCP flows window size are not limited by small receiver's advertised window sizes. Extensions to include this situation is straightforward but details are skipped for space consideration. Finally, **A4** allows us to model only "unforced" RED packet losses [1] and ignore forced losses which should be avoided when configuring RED parameters since they cause the RED gateway to act as a Tail Drop gateway. Hence, we assume that the RED gateway parameters are set such that the number of forced packet losses is negligible. [2] Section III presents an analytic derivation of the required RED parameters for achieving this goal. The analytical results are supported by *ns* simulation for a wide range of parameters.

III. RED Analysis with TCP Flows

A. Analysis

Consider the scenario of Fig 1 with M TCP flows; let Z_i denote the time at which the i^{th} random packet loss (caused by RED) event takes place, and $X_i = Z_i - Z_{i-1}$ denote the inter-loss duration (*epoch*). Let the j^{th} TCP session window size (in packets) at the beginning and end of the i^{th} epoch be denoted by $W_{i,j}$ and $W_{i+1,j}$ respectively. Let $Q_{i,j}$ denote the queue size for session j packets at the end of epoch i; then the net round-trip propagation and queuing delay for connection j at the end of the i^{th} epoch is $\gamma_{i,j} = T_j + Q_{i,j}/\mu$. We postulate the probability $P_{i,j}$ that the i^{th} packet loss event belongs to the j^{th} flow is given by the share (fraction) of the flows present bandwidth, i.e.,

$$P_{i,j} = \frac{(\frac{W_{i,j}}{\gamma_{i,j}})}{\sum_{k=1}^{M}(\frac{W_{i,k}}{\gamma_{i,k}})} \tag{2}$$

For multiple Poisson processes (flows) sharing a common gateway that randomly drops packets with a fixed drop proba-

[1] Forced packet losses are those due to either buffer overflow or due to packet drop with probability one.

[2] This assumption has also been adopted in almost all previous TCP-related research efforts, whether with RED or with Tail Drop gateways (e.g. [13], [14], [16], [17], [20], [24]) with (up to our knowledge) the exception of [11]).

bility, it is well known that the probability that a dropped packet belongs to a flow is exactly equal to the ratio of the corresponding arrival rate to the net (total) arrival rates of all processes (e.g. see [25] pp.41). Equation 2 is a reformulation of this result applied to the case of RED gateways with TCP flows, where the current flow rate is approximated by[3] the ratio of the current window size to the round-trip time.

Since all flows share the same buffer, $Q_{i,j} = Q_i \forall j$. Let $E[Q_i] = \overline{Q}$ denote the steady-state average queue size. For a congested RED gateway, the average queue size $\overline{Q} \in [min_{th}, max_{th}]$, with corresponding random drop probability between $[0, max_p]$. Note that the inter-loss durations X_i while identically distributed with mean $\overline{X}$ are, strictly, not independent.

Based on the preceding discussion and the assumptions of Section II, the window evolution of the TCP-Reno sessions is governed by the following system of equations (refer to Figure 2),

$$W_{i+1,j} = \begin{cases} W_{i,j} + \frac{X_i}{\gamma_{i,j}} & \text{w.p.} \quad 1 - P_{i,j} \\ W_{i,j}/2 + \frac{X_i}{\gamma_{i,j}} & \text{w.p.} \quad P_{i,j} \end{cases} \tag{3}$$

Denote $E[\gamma_{i,j}] = \overline{\gamma_j} = T_j + \frac{\overline{Q}}{\mu}$. Taking the expectation of both sides of (3)

$$\frac{\overline{W_j}\,\overline{P_j}}{2} = \frac{\overline{X}}{\overline{\gamma_j}} \quad j = 1, \ldots M \tag{4}$$

where we used the independence approximation (see Section II) $E[W_{i,j}\, P_{i,j}] = \overline{W}_j\, \overline{P}_j$ and a fixed point approximation.

Denote $\widetilde{W}_j = \frac{\overline{W_j}}{\overline{\gamma_j}}$, then, by fixed point approximation to (2),

$$\overline{P_j} = \frac{\widetilde{W}_j}{\sum_{k=1}^{M} \widetilde{W}_j}. \tag{5}$$

Substituting by (5) in (4),

$$\overline{W_j} = \sqrt{2\overline{X} \sum_{k=1}^{M} \widetilde{W}_k} \quad j = 1, \ldots, M \tag{6}$$

Equation (6) is a system of M quadratic equations in the M unknowns $\overline{W}_j$ (or $\widetilde{W}_j$). Denoting,

$$\alpha_j = \frac{\overline{X}}{\overline{\gamma_j}^2} \tag{7}$$

it can be readily verified by direct substitution that

$$\widetilde{W}_j = 2\sqrt{\alpha_j} \sum_{k=1}^{M} \sqrt{\alpha_k} \tag{8}$$

[3]The approximation of the TCP flow rate as the ratio of the window size to the round-trip time has been used in many TCP modeling approaches and has shown to be reasonably accurate.

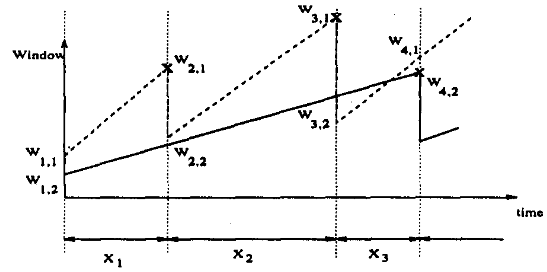

Fig. 2. A schematic showing the congestion window size for two (i.e. $M = 2$) TCP-Reno sessions with different round-trip delays sharing a bottleneck link with RED gateway. Packet loss events are indicated by an X.

is an explicit solution for (6) and hence,

$$\overline{W_j} = 2\overline{X} \sum_{k=1}^{M} \frac{1}{\overline{\gamma_j}} \tag{9}$$

Remarkably, the above result implies that *the steady-state window size is the same for all flows*; also, $\overline{W_j}$ implicitly depends on $\overline{Q}$ and $\overline{X}$ that need to be determined.

Let $\overline{W_{av_j}}$ denote the *time-averaged* mean window size of session j during an epoch, as opposed to $\overline{W}_j$ that represents the mean window size just prior to end of the epoch; thus $\overline{W_{av_j}} = K\overline{W}_j$, where $K < 1$. Hence,

$$\overline{R_j} = \frac{\overline{W_{av_j}}}{\overline{\gamma_j}} = K\frac{\overline{W_j}}{\overline{\gamma_j}} = \frac{2\overline{X}K}{\overline{\gamma_j}} \sum_{k=1}^{M} \frac{1}{\overline{\gamma_j}} \tag{10}$$

In case of full bottleneck link utilization,

$$\sum_{j=1}^{M} \overline{R_j} = \mu \tag{11}$$

which, by substituting from (10) and (9) yields,

$$\mu = K \sum_{j=1}^{M} \frac{\overline{W_j}}{\overline{\gamma_j}} = 2K\overline{X} \left(\sum_{j=1}^{M} \frac{1}{\overline{\gamma_j}} \right)^2 \tag{12}$$

Finally, let $\overline{p}$ denote the average packet drop probability of the RED gateway. Then,

$$\overline{p} = \frac{1}{\mu \overline{X}} \tag{13}$$

Since the RED average packet drop probability also satisfies

$$\overline{p} = \frac{\overline{Q} - min_{th}}{max_{th} - min_{th}}\, max_p \tag{14}$$

a relationship between $\overline{Q}$ and $\overline{X}$ follows by equating (13) and (14).

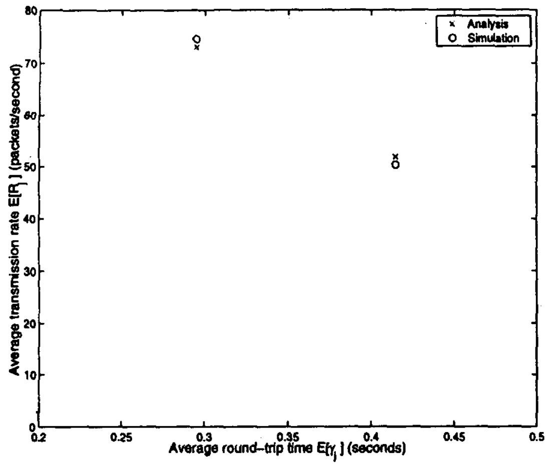

Fig. 3. Results for RED gateway with 2 flows - see parameters in Table I.

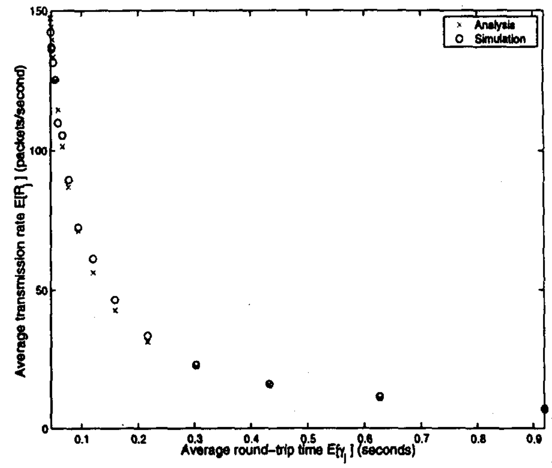

Fig. 4. Results for RED gateway with 16 flows - see parameters in Table 1.

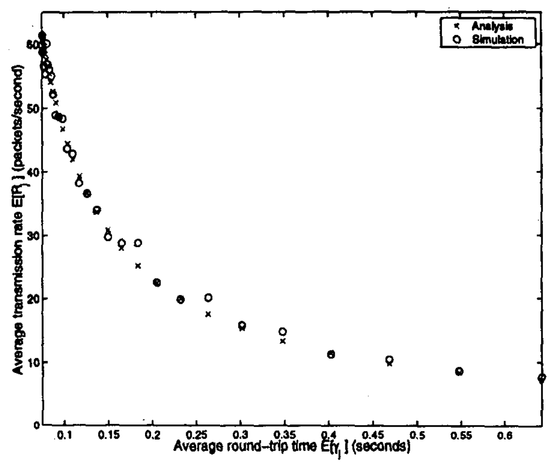

Fig. 5. Results for RED gateway with 32 flows - see parameters in Table 1.

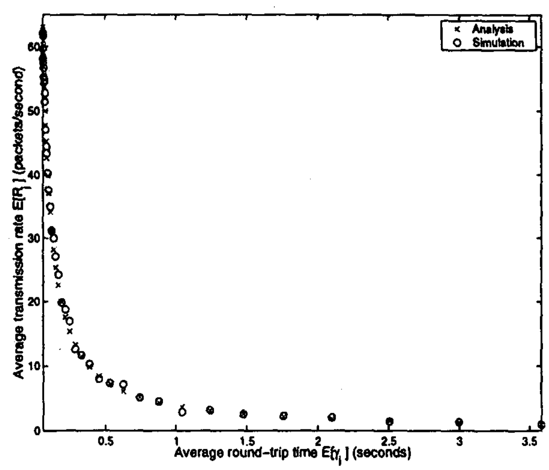

Fig. 6. Results for RED gateway for the first 42 flows of the 64 flows simulation - see parameters in Table 1.

B. Simulation Results

In the preceding subsection, we derived analytic expressions for various parameters of interest for RED gateways using a number of well-motivated approximations. In this subsection, we support and complement the analytical results by comparing with simulations.

Table I lists the parameters used for four sets of simulation experiments with RED policy using the *ns* simulator [21]; the link capacities μ and μ_j (packets/second), propagation delays τ and τ_j (seconds) for the topology in Fig. 1 as well as the number of competing flows used for each simulation. In each simulation, a value for the link delay τ_1 of the first flow and a factor $f > 1$ is chosen such that the remaining τ_j are specified according to $\tau_j = f * \tau_{j-1}$, i.e. the propagation delay profile increases exponentially. The values of τ_1 and f for each simulation are listed in Table 1. For each simulation we compare the analysis results to the simulations measurements of $\overline{Q}$ in Table 1. All simulations use equal packet size of 1 KBytes. The factor K is found to be equal to 0.5 for all experiments. Figures 3-7 show the measured and the computed $\overline{R_j}$ for each competing flow, where the x-axis indicates the computed $\overline{\gamma_j}$. We choose to show $\overline{R_j}$ in packets/second rather than the normalized throughput since the difference between the simulations measurement and the analysis results are indistinguishable if shown normalized to the link capacity.

C. On the Fairness of RED

It is well known that TCP is inherently biased towards flows with shorter round-trip time, mainly due to its window increase mechanism. In [12], it has been shown that, in case of multiple TCP flows with different round-trip times sharing a Tail Drop gateway, the steady-state throughput of a flow is inversely proportional to the *square* of the average round-trip delay.

The analysis/simulations results of RED in this paper highlight a dependence of the throughput of the competing TCP

TABLE I

RED GATEWAY PARAMETERS SETTING FOR EACH SIMULATION

M	μ	τ	μ_j	τ_1	f	min_{th}	max_{th}	$1/max_p$	Q (an.)	Q (sim.)
2	125	0.002	1250	0.02	4	20	80	80	30.3467	24.9884
16	1250	0.005	1250	0.001	1.5	10	100	16	41.4134	47.8644
32	1250	0.005	1250	0.001	1.2	10	100	16	77.6440	82.2968
64	1250	0.005	1250	0.001	1.2	10	100	8	59.5863	59.4377

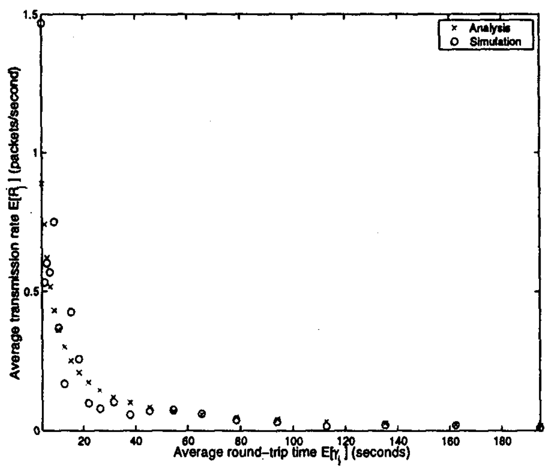

Fig. 7. Results for RED gateway for the last 22 flows of the 64 flows simulation - see parameters in Table 1.

connections that differs from that observed for Tail Drop. *The ratio of the bandwidths of the competing connections is not inversely proportional to the square of the ratio of the average round-trip times, but is inversely proportional only to the ratio of the average round-trip times.*

Thus, if we define an internal fairness coefficient ρ as the ratio of the lowest to the highest average throughput among the competing flows sharing a gateway (i.e. ideally, we would like $\rho = 1$), then, comparing the fairness coefficients of a RED gateway to an equivalent Tail Drop gateway[4], the RED gateway results in a fairness coefficient that is closer to 1. Specifically, if ρ_{TD} is the fairness coefficient of the Tail Drop gateway, then the fairness coefficient of an equivalent RED gateway is $\sqrt{\rho_{TD}}$.

An intuitive explanation for this decrease in bias (i.e. fairness coefficient closer to one) for RED is in order. TCP increases its window size by one every round-trip time. Thus, lower delay links increase their window faster. In case of synchronized packet loss (Tail Drop gateways), all windows are (typically) reduced at the same time, and hence the average window size is inversely proportional to the average round-trip time. But, the transmission rate is proportional to the window size divided by the round-trip time. Hence, the average rate of each TCP session is inversely proportional to the square of the average round-trip time.

In case of RED however, conditioned on an arriving packet being dropped, the chances that the packet will belong to a certain connection is (on the average) proportional to that connection transmission rate. Thus TCP sessions with lower delays are more likely to have their windows reduced. The analytical results show that this causes the *average window size* to be *equal* for all connections (6) and thus the throughput of a connection is only inversely proportional to the round-trip-delay, and *not* the square of the round-trip delay as in the case of Tail Drop. Thus while the basic RED algorithm does *not* achieve throughput fairness among the competing flows, it substantially reduces the bias as compared to Tail Drop.

D. *A lower bound on $\overline{Q}$ for a RED gateway with no deterministic drops*

While the structure of the RED algorithm is well understood, little has been published regarding the possibility of achieving certain design objectives using RED gateways. In this subsection, we consider a RED gateway to a bottleneck link with known input TCP traffic, and wish to find a lower bound on the minimum achievable $\overline{Q}$ using a RED gateway that satisfies the RED Control Condition (i.e. no deterministic drops) described in Section III.

Let us assume (in order to achieve closed form results) that the input traffic is composed of TCP flows with equal delays $T_j = T\ \forall j$. Define the *normalized* number of flows r as

$$r = \frac{M}{\mu T} \tag{15}$$

thus, from (12) and (13),

$$\overline{p} = (\frac{r}{1 + \frac{\overline{Q}}{\mu T}})^2 \tag{16}$$

Consider the two cases $r < 1$ and $r > 1$. The former case describes a traffic input where the number of flows M is less than the link bandwidth-delay product, while in the latter there are more flows than packets. Let $\overline{Q}_{min}$ denote the lower bound on $\overline{Q}$; then, since $\overline{p} \leq 1$, from (16)

$$\overline{Q}_{min} = \begin{cases} 0 & M \leq \mu T \\ M - \mu T & M > \mu T \end{cases} \tag{17}$$

[4]An *equivalent* gateway is one that has identical TCP flows and the same average queue size ([4])

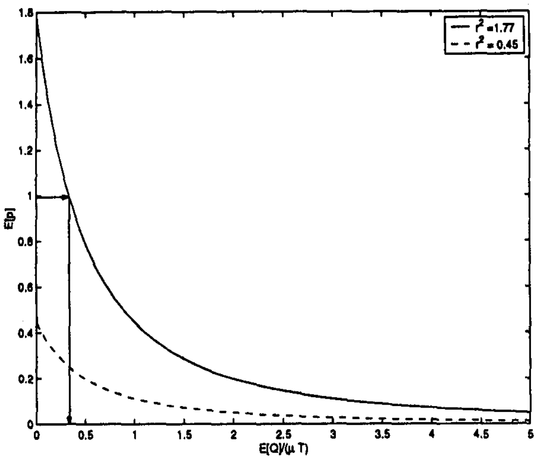

Fig. 8. Lower bounds on achievable $\overline{Q}$ using a RED gateway.

Figure 8 shows a numerical example for the two cases of r, where we plot the average drop probability as a function of the *normalized* queue size for different values of the normalized number of flows r.

The above bound on $\overline{Q}$ simply states the following: if the number of identical competing flows is greater than the bandwidth-delay product ($r > 1$), then it is impossible to configure the parameters of a RED gateway to achieve an average queue size for a given input traffic that is less than $\overline{Q}_{min} = M - \mu T$ without having to deterministically drop packets (i.e. either drop packets w.p. 1 or drop packets due to buffer overflow). On the other hand, if $r \leq 1$, then there exists a RED gateway that achieves any $0 < \overline{Q_o} < \infty$ (where $\overline{Q_o}$ is a desired operating average queue size), but the corresponding $\overline{p_o}$ must lie on the $\overline{p}$-$\overline{Q}$ curve above, which implies that the parameters of the RED gateway must be chosen such that the straight line between $(min_{th}, 0)$ and (max_{th}, max_p) passes through the operating point $(\overline{Q_o}, \overline{p_o})$.

IV. Conclusion

In this paper, we have developed a simple analytical model for a RED gateway with multiple competing TCP flows. As an application of the model, we studied the impact of a congested RED gateway on the inherent TCP bias against higher delay links. We found that the fairness (defined as the ratio of the lowest to the highest average throughput among a set of competing TCP flows) of a RED gateway (with no deterministic drops) is square-root of that of an equivalent Tail Drop gateway.

Our analytical results for RED were used to derive a simple bound on the values of the average queue size and packet loss rate achievable by a RED gateway with no deterministic drops.

A number of avenues for future extensions remain. First, our model can be enhanced to account for the effects of time-outs. Second, a more precise calculation can be obtained by modeling the window size as a Markov Chain (similar to the work in [11]). Third, the model can be further extended to incorporate the effects of smaller windows and deterministic drops due to limited buffer size. Finally, the model can be extended to other versions of TCP (e.g. TCP Tahoe).

References

[1] L. Zhang and D. Clark, "Oscillating behavior of network traffic: a case study simulation," *Internetworking: Research and Experience*, vol. 1, no. 2, pp. 101–12, 1990.

[2] V. Jacobson, "Presentation at the IETF meeting," in *Proceedings of the Sixth Internet Engineering Task Force*, 1987.

[3] E. Hashem, "Analysis of random drop for gateway congestion control," Report LCS TR-46, Laboratory for Computer Science, MIT, Cambridge, MA, 1989.

[4] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, 1993.

[5] S. Floyd, "TCP and explicit congestion notification," *ACM Computer Communication Review*, vol. 24, no. 5, pp. 10–23, 1994.

[6] K. Ramakrishnan and S. Floyd, "A proposal to add explicit congestion notification (ECN) to IP," Tech. Rep. RFC 2481, IETF, January 1999.

[7] W. Feng, D. Kandlur, D. Saha, and K. Shin, "Techniques for eliminating packet loss in congested TCP/IP networks," Technical Report CSE-TR-349-97, University of Michigan, 1997.

[8] W. Feng, D. Kandlur, D. Saha, and K. Shin, "A self-configuring RED gateway," in *Proceedings of INFOCOM'99*, 1999.

[9] B. Braden et al, "Recommendations on queue management and congestion avoidance in the Internet," Informational RFC2309, IETF, April 1998.

[10] J. Bolot, "End-to-end packet delay and loss behavior in the Internet," in *Proceedings of ACM SIGCOMM'93*, 1993, pp. 289–298.

[11] A. Abouzeid, S. Roy, and M. Azizoglu, "Stochastic modeling of TCP over lossy links," in *Proceedings of INFOCOM'2000*, 2000.

[12] O. Ait-Hellal, *Flow Control in High-Speed Networks*, Ph.D. thesis, Universite de Nice-Sophia Antiopolis, November 1998.

[13] M. Mathis, J. Semke, J. Madhavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *Computer Communications Review*, vol. 27, no. 3, pp. 67–82, 1997.

[14] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," *Computer Communication Review*, vol. 28, no. 4, pp. 303–314, 1998.

[15] P. Brown, "Resource sharing of TCP connections with different round trip times," in *Proceedings of INFOCOM'2000*, 2000.

[16] S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the internet," *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 458–472, 1999.

[17] V. Firoiu and M. Borden, "A study of active queue management for congestion control," in *Proceedings of INFOCOM'2000*, 2000.

[18] T. Bonald, M. May, and J-C. Bolot, "Analytic evaluation of RED performance," in *Proceedings of INFOCOM'2000*, 2000.

[19] A. Mokhtar and M. Azizoglu, "A random early discard framework for congestion control in ATM networks," in *Proceedings of IEEE ATM Workshop'99*, 1999, pp. 45–50.

[20] V. Misra, W-B. Gong, and D. Towsley, "Fluid-based analysis of a network of aqm routers supporting tcp flows with an application to red," in *Proceedings of SIGCOMM'2000*, 2000.

[21] S. McCanne and S. Floyd, "ns-Network Simulator," *URL http://www-mash.cs.berkeley.edu/ns/*, 1995.

[22] F. P. Kelly, "Blocking probabilities in large circuit-switched networks," *Advances in Applied Probability*, vol. 18, pp. 473–505, 1986.

[23] T. Lakshman and U. Madhow, "The performance of TCP/IP for networks with high bandwidth-delay products and random loss," *IEEE/ACM Transactions on Networking*, vol. 5, no. 3, pp. 336–50, 1997.

[24] S. Savari and E. Telatar, "The behavior of certain stochastic processes arising in window protocols," 1999, pp. 791–5.

[25] R. Gallager, *Discrete Stochastic Processes*, Boston : Kluwer, 1996.