

Beyond CHOKe: Stateless Fair Queueing

Rade Stanojević and Robert Shorten

Hamilton Institute, NUIM, Ireland

Abstract. Making drop decisions to enforce the max-min fair resource allocation in a network of standard TCP flows without any explicit state information is a challenging problem. Here we propose a solution to this problem by developing a suite of stateless queue management schemes that we refer to as Multi-Level Comparison with index l ($MLC(l)$). We show analytically, using a Markov chain model, that for an arbitrary network topology of standard TCP flows and queues employing $MLC(l)$, the resource allocation converges to max-min fair as l increases. The analytical findings are verified experimentally using packet level *ns2* simulations.

1 Introduction

Resource allocation in communication networks has been a topic of interest for some time. In the current Internet most traffic uses TCP as the transport protocol, and most Internet routers do not differentiate packets from different flows. In order to adjust the resource allocation amongst competing users, one can do the following: (1) design the new end-to-end protocol(s) and leave the network infrastructure (routers) unchanged [28, 12, 8]; (2) design the new end-to-end protocol(s) and network support that will allow cooperation between end-users and network (routers) [11, 3, 27]; (3) leave the end-to-end protocol(s) unchanged but design the network based scheme that determines desired resource allocation [25, 5, 22].

Most current proposals have as their performance objective a resource allocation that is max-min fair. In this paper we propose a scheme that belongs to the third group listed above, and whose performance goal is enforcing a max-min fair resource allocation. Its main features are the following.

1. No changes to end-to-end transport protocols are required.
2. The decision to drop (mark) a packet is made locally by each router;
3. No multiple queues or per flow counters are used.

Thus, our goal is to design a stateless active queue management scheme that can enforce max-min fairness in the network of TCP users. While there exists a large amount of work related to analysis and design of distributed algorithms/architecture that enforce max-min fairness, to the best of our knowledge, our algorithm is the first that attempts a stateless active queue management scheme to enforces max-min fairness in the network of TCP users.

1.1 Paper contributions

Why is reaching the goal stated above so hard? First, recall that in the max-min fair regime, a TCP flow f experiences drops at *one and only one* link l_f at its path (we say that f is bottlenecked at l_f), and therefore must be protected at other links (that can be congested) by receiving lossless service. Second, if two or more flows are bottlenecked at the same link they must receive nonuniform loss rates that are function of their aggressiveness (round-trip times, queuing delays, delayed acks option, etc). Assuming that the router has access to the individual flow rates, or the existence of multiple queues that are appropriately scheduled, a number of solutions to these two problems exist, and are described in previous works [25, 11, 20, 5, 22]. However, in our case (routers with no explicit state information) it is highly nontrivial to make the drop (mark) decision without any explicit information. The main contributions of this paper are:

- A stateless queue management scheme, Multi Level Comparisons with index l (MLC(l)), which makes the drop decision based on the structure of packets that are already in the queue (using simple comparisons only).
- A Markov chain analysis of the randomized algorithm MLC(l) that shows that the resource allocation of MLC(l) converge to the max-min fair, for arbitrary network topology and arbitrary set of TCP users, as the index l grows.
- Packet level simulations are presented that support the analytical findings.

2 Power-drop AQM schemes

It has been noticed in many studies that both drop tail and RED routers have large bias against large-RTT flows. For example, the authors of [14] have made the empirical observation that for a drop-tail router and two flows with round trip times RTT_1 and RTT_2 , the ratio of asymptotic throughput of the first and the second flow is in the ratio $(RTT_2/RTT_1)^a$ for some $a \in (1, 2)$. Similarly, it has also been noticed in a number of studies, that oblivious (ones that do not differentiate packets from different flows) AQM schemes (RED [10], BLUE [7], etc.) which attempt to estimate the loss probability for a given traffic pattern and to drop packets according to this estimation, share bandwidth among competing users with round trip times RTT_1 and RTT_2 in the ratio RTT_2/RTT_1 , [9, 2]. In this section we will investigate RTT unfairness characteristics for more general AQM schemes we call power-drop AQM schemes.

Definition 1. *An AQM is power-drop if it drops a packet from a flow with current throughput¹ U with probability $\rho_0 U^{l-1}$, where ρ_0 is variable controlled by router and l positive integer called index of the given power-drop AQM.*

¹ Throughput is measured in packets per unit of time.

Comment: With $l = 1$ this corresponds to a router which drops packets with loss probability ρ_0 . The case when $l = 2$ is similar to CHOKe[21] in the limit when the average queue size does not go below minimum threshold, and in addition, when there is neither a RED nor an overflow drop. Indeed, comparing a packet at the entrance of queue with a packet from the queue and making drop-decision based on this comparison is actually dropping a packet with probability which is proportional to current throughput of the flow.

In this section we will describe a class of power-drop AQM's called Multi Level Comparison (MLC) AQM's. In particular, we will describe and analyze the fairness characteristics of this queueing discipline for TCP flows competing for bandwidth. We will see that the MLC scheme with index l achieves $1/RTT^{1/(l+1)}$ -fairness² under the assumption of low loss probability (Theorem 1). More generally, Theorem 2 shows that increasing l leads resource allocation among TCP users arbitrarily close to max-min fairness in general network topologies.

2.1 Description of MLC

The basic strategy in $MLC(l)$ is to extend the core idea from CHOKe of comparing of a packet arriving at the queue with packets which are already in queue; these stored packets are measure of the proportion of bandwidth used by certain flow. $MLC(l)$ maintains a variable h_M which is used to control the probability of dropping an arriving packet: at every packet arrival h_M dropping trials are executed.

Dropping trial: Pick randomly $l - 1$ packets from the queue: if *all* l packets belong to same flow, then drop the arriving packet (if $l = 1$ the arriving packet is dropped by default).

If the arriving packet is not dropped after the execution of h_M dropping trials then it is enqueued. If h_M is not an integer, the number of dropping trials is given as follows. For $h_M < 1$ we execute 1 dropping trial with probability h_M and 0 dropping trial with probability $1 - h_M$. Similarly, $h_M > 1$ we execute $\lfloor h_M \rfloor + 1$ dropping trials with probability $\{h_M\} = h_M - \lfloor h_M \rfloor$ and $\lfloor h_M \rfloor$ dropping trials with probability $1 - \{h_M\}$.

Proposition 1. *For a given h_M , $MLC(l)$ is power-drop scheme with index l .*

Proof. Let U_f be the throughput of a flow f , and U_0 the aggregate throughput on the link. A packet is dropped at one dropping trial with probability

$$q_1 = \left(\frac{U_f}{U_0} \right)^{l-1}.$$

² Two flows with round trip times RTT_1 and RTT_2 which have a single bottleneck operating with MLC, obtain bandwidth in ratio: $(RTT_1/RTT_2)^{-1/(l+1)}$.

The probability that a packet is dropped after h_M trials is $1-Prob[\text{packet is not dropped at any of } h_M \text{ trials}]$ which is given by

$$q = 1 - (1 - q_1)^{h_M} \approx q_1 \cdot h_M = U_f^{l-1} \frac{h_M}{U_0^{l-1}}.$$

Taking $\rho_0 = \frac{h_M}{U_0^{l-1}}$ we conclude that $MLC(l)$ satisfies Definition 1.

The higher h_M the more frequent the losses are. Consequently, if the link is under-utilized h_M should be decreased in order to decrease probability of dropping packets. On the other hand if the aggregate traffic on the link is greater than the link capacity then h_M should be increased to reduce traffic load.

Controlling the variable h_M : MLC uses a parameter Δ_0 to affect changes in the variable h_M (in our simulations Δ_0 is set to $100ms$). h_M is adjusted once per Δ_0 using a MIMD (Multiplicative Increase - Multiplicative Decrease) scheme. The performance goal is to keep the utilization at a certain level u_0 . Namely, if within the previous Δ_0 the link utilization was less than desired u_0 , h_M is set to h_M/γ for some $\gamma > 1$, otherwise h_M is adjusted as $h_M := h_M\gamma$.

At this point it is important to emphasize a few differences between MLC and CHOKe. First, note that CHOKe makes a comparison only when the average queue size becomes greater than min_{th} (RED minimum threshold), and therefore its performance (in terms of resource allocation between TCP users) depends mainly on the number of users: a small number of TCP flows will affect the synchronization of losses, while for large number of users, the number of CHOKe-drops will be much less than number of RED-drops and therefore the effect of CHOKe on TCP fairness would be negligible. Second, the design of CHOKe basically neglects the TCP fairness as performance objective and concentrates on reducing throughput of unresponsive flow(s) [26], while MLC is designed to improve TCP fairness and neglects effects of unresponsive flows.

Our experiments indicate that the parameter Δ_0 should be in the range of round trip times of the connections using the link (in order to allow users to react to changes in h_M). The parameter γ controls the speed of adaptation to changes in network traffic and should be set such that, for a given Δ_0 , h_M can be doubled/halved within a few seconds.

2.2 Model and analysis of power-drop AQM

In this section we present a model of power-drop AQM's servicing multiple TCP users. We present results that characterize this situation for both a single bottleneck and for general network topologies.

Single bottleneck case: We consider N TCP-flows with heterogenous round trip times RTT_i , $i = 1, \dots, N$, traversing a single bottleneck link that employs power-drop AQM with an index l . If we assume that ρ_0 does not fluctuate much (so that we can model it as constant) and that the drop probability for a packet is small, then our analysis shows that the asymptotic rates achieved by

TCP users are proportional to $\frac{1}{RTT_i^{2/(l+1)}}$. This is the main result of this section and is given in Theorem 1.

Model : At the flow level, let Δ be the length of sampling interval over which we evaluate changes in throughput. If a flow with a round trip time RTT does not see a drop within interval of length Δ , then its throughput will be increased for Δ/RTT^2 (see [1]). If a flow registers a drop within this sampling interval then its throughput will be halved³. The probability that the first event will happen is equal to the probability that each of ΔU packets from the flow are not dropped. This probability is given by:

$$\eta_1 = (1 - \rho_0 U^{l-1})^{\Delta U} \approx e^{-\Delta \rho_0 U^l}.$$

Clearly, the probability that a flow with current throughput U will see a drop within a sampling interval of length Δ is equal to

$$\eta_2 = 1 - (1 - \rho_0 U^{l-1})^{\Delta U} \approx 1 - e^{-\Delta \rho_0 U^l}.$$

The previous approximations are valid under the assumption of a small probability that a packet will be dropped : $\rho_0 U^{l-1} \ll 1$. This assumption seems reasonable, since if this probability is not small, a flow would suffer too many losses and therefore would not get chance to enter the (AIMD) congestion avoidance phase.

Let $U_k^{(\rho)}$ be a stochastic process which describes the evolution of throughput of a TCP flow with round-trip time RTT traversing over link with a power-drop AQM scheme with index l . Here $\rho = \Delta \rho_0$. Since Δ is fixed we can assume that Δ is equal to one unit of time.

We model $U_k^{(\rho)}$ as a Markov chain on $[0, \infty)$ defined by $U_0^{(\rho)} = 0$ and:

$$U_{k+1}^{(\rho)} = U_k^{(\rho)} + \frac{1}{RTT^2} \quad \text{with probability } e^{-\rho(U_k^{(\rho)})^l}$$

$$U_{k+1}^{(\rho)} = \frac{1}{2} U_k^{(\rho)} \quad \text{with probability } 1 - e^{-\rho(U_k^{(\rho)})^l}.$$

The following theorem characterizes the time averaged throughput of a TCP flow with round trip time given by RTT , running over power-drop queue management scheme with index l and its proof can be found in [24].

Theorem 1. *The time averaged throughput of the i 'th flow: $\frac{1}{M} \sum_{i=1}^M U_i^{(\rho)}$ converges almost surely to:*

$$\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{i=1}^M U_i^{(\rho)} =: \bar{U}^{(\rho)} = \frac{1}{RTT^{\frac{2}{l+1}} \rho^{\frac{1}{l+1}}} D_{MLC}(l) + \frac{1}{\rho^{\frac{1}{l+1}}} S^{(\rho)}$$

where $D_{MLC}(l)$ is a constant that does not depend on ρ neither RTT and $S^{(\rho)}$ converges to 0 as ρ goes to 0.

³ Throughout this paper, variations in round trip times are neglected.

Remark: The previous theorem is a generalization of the well known square root formula. Indeed, for $l = 1$, power-drop scheme is an oblivious AQM that drops packets with probability $\rho = h_M$ and Theorem 1 says that time averaged throughput converge to $\frac{1}{RTT\sqrt{\rho}}D_{MLC}(1) + o(\frac{1}{\sqrt{\rho}})$.

To conclude this section we prove that for a given network with routers employing a power-drop AQM with index l , and assuming that the steady state throughput is given by the previous theorem, we can find large enough l such that bandwidth allocation is arbitrary close to max-min fairness. The following characterization of max-min fairness can be found in [23].

Lemma 1. *A set of rates x_r is max-min fair if and only if for every flow r there exists a link on its path, such that the rates of all flows which traverse through that link are less or equal than x_r .*

With this characterization of max-min fair allocation in mind, we shall prove that increasing the index of the MLC will result in allocation of bandwidth in such fashion that each flow will have link on its path such that its asymptotic rate is “almost” the largest among all flows using that link.

Theorem 2. *For any given network topology, and given $\varepsilon > 0$, there exists l such that if all queues employ MLC with index l and loss probabilities are small then for every flow r there exist a link on its path, such that the rates of all flows which traverse through that link are less than $(1 + \varepsilon)x_r$ (here x_r is steady state rate of flow r).*

Proof. Let L be the number of links in the network and N the number of flows. We label flows by $i = 1, 2, \dots, N$ and links by $s = 1, 2, \dots, L$. By R we denote the routing matrix: $R_{is} = 1$ if flow i uses link s otherwise $R_{is} = 0$. On each link s , a router drops a packet from the flow with current throughput U with probability $\rho(s)U^{l-1}$. Let M be the length (in number of links) of the path of the flow with most links on its route and ν the ratio of the largest and the smallest round trip time in the network. Choose l such that

$$\nu^{\frac{2}{l+1}} M^{\frac{1}{l+1}} < 1 + \varepsilon.$$

For each flow r , let $s_1^{(r)}, \dots, s_w^{(r)}$ be links used by it and let $s_{max}^{(r)}$ the most congested link on its route in the following sense:

$$\rho(s_{max}^{(r)}) = \max\{\rho(s_j^{(r)}) \mid j = 1, \dots, w\}. \quad (1)$$

If the current rate of flow r is U , a packet from that flow will be dropped with probability $\lambda_r U^{l-1}$, where $\lambda_r = \sum_{j=1}^w \rho(s_j^{(r)})$, and therefore the steady state throughput for flow r is given by

$$x_r = \frac{1}{RTT_r^{\frac{2}{l+1}} \lambda_r^{\frac{1}{l+1}}} C_0.$$

For any other flow t which uses the link $s_{max}^{(r)}$ with $\lambda_t = \sum_{j:R_{jt}=1} \rho(l_j) \geq \rho(s_{max}^{(r)})$ the steady state throughput is given by:

$$x_t = \frac{1}{RTT_t^{\frac{2}{l+1}} \lambda_t^{\frac{1}{l+1}}} C_0.$$

Recall that we have defined the link $s_{max}^{(r)}$ as the most congested link on route of flow r in the sense of (1). This implies that $\lambda_r \leq M\rho(s_{max}^{(r)})$. Now

$$\begin{aligned} \frac{x_t}{x_r} &= \left(\frac{RTT_r}{RTT_t} \right)^{\frac{2}{l+1}} \left(\frac{\lambda_r}{\lambda_t} \right) \leq \left(\frac{RTT_r}{RTT_t} \right)^{\frac{2}{l+1}} \left(\frac{M\rho(s_{max}^{(r)})}{\rho(s_{max}^{(r)})} \right)^{\frac{1}{l+1}} \leq \\ &\leq \nu^{\frac{2}{l+1}} M^{\frac{1}{l+1}} < 1 + \varepsilon. \end{aligned}$$

Remark: Note that for a single bottleneck topology, the resource allocation given by $\frac{C}{RTT_i^{2/(l+1)}}$ is $((RTT_i^2), l+1)$ proportionally fair[18, 23]. Indeed, for any resource allocation (x_i) , utility $U(x) = \sum_{i=1}^N \frac{RTT_i^2}{x_i^l}$, and link capacity c_0 we have (using Holder's inequality):

$$\begin{aligned} (U(x))^{\frac{1}{l+1}} \cdot c_0^{\frac{l}{l+1}} &= \left(\sum_{i=1}^N \frac{RTT_i^2}{x_i^l} \right) \cdot \left(\sum_{i=1}^N x_i \right) = \\ &\left(\sum_{i=1}^N \left(\frac{RTT_i^{2/(l+1)}}{x_i^{\frac{l}{l+1}}} \right)^{l+1} \right)^{\frac{1}{l+1}} \cdot \left(\sum_{i=1}^N \left(x_i^{\frac{l}{l+1}} \right)^{\frac{l+1}{l}} \right)^{\frac{1}{l+1}} \leq \\ &\leq \sum_{i=1}^N \frac{RTT_i^{2/(l+1)}}{x_i^{\frac{l}{l+1}}} \cdot x_i^{\frac{l}{l+1}} = \sum_{i=1}^N RTT_i^{2/(l+1)} \end{aligned}$$

$U(x)$ is maximized if equality holds in the inequality above, which is equivalent to $x_i = \frac{C}{RTT_i^{2/(l+1)}}$ for some constant C . Thus, while spectrum of delay-based end-to-end protocols[18] assume no cooperation from routers to converge to max-min fairness, MLC(l) does not require changes in end-to-end protocol to enforce max-min fairness.

Similar interesting feature is shared between shuffling parameter $\gamma > 0$ of XCP and index l of MLC, but is not discussed here because of space limitations; see [24, 16].

3 Experimental results

In this section we briefly describe some *ns2* simulations that demonstrate the behavior of proposed AQM schemes.

3.1 Single bottleneck

The first set of simulations are designed to demonstrate the fairness properties of the MLC in single bottleneck scenario. Specifically, we present results for a single link with service rate of $80Mbps$ that services 100 long-lived TCP users with round trip times uniformly distributed in range $40 - 440ms$. To provide baseline results, we include the performance of RED for the same scenario. Share of total throughput taken by each of 100 flows assuming the bottleneck queue is managed by MLC(2) is depicted in Figure 1.

It can be seen from Figure 1 that the fairness of RED is approximately proportional to the inverse of RTT. This is in accordance with observations made in [2, 9]. It can also be observed that the fairness of MLC with index 2 is proportional to $1/RTT^{2/3}$ as predicted by Theorem 1. The MLC parameters used in the simulation are: $l = 2$, $\Delta_0 = 100ms$, $\gamma = 1.01$, $u_0 = 0.98$.

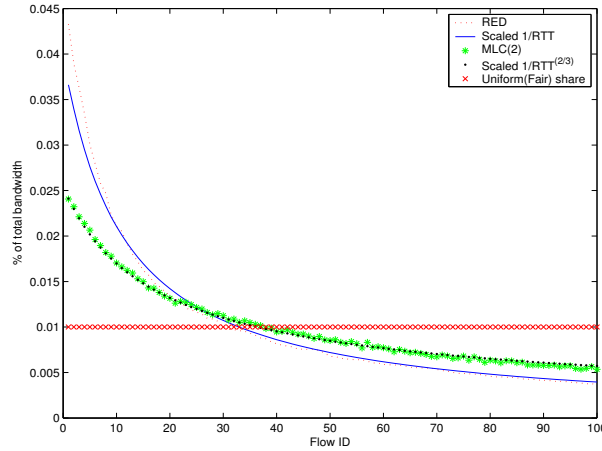


Fig. 1. Scaled throughput for 100 flows over congested link employing RED, MLC(2).

3.2 Multiple bottleneck topologies

Our second set of simulations demonstrate Theorem 2. The network topology that we considered is given in Figure 2. Here, we consider a network of 24 nodes: $n1 - n5, m1 - m5, p1 - p5, q1 - q5$, and $c1, c2, c3, c4$ and 30 flows traversing the network as follows: $n(i) \rightarrow p(i); n(i) \rightarrow q(i), m(i) \rightarrow p(i); m(i) \rightarrow q(i); n(i) \rightarrow m(i); p(i) \rightarrow q(i)$ where $i = 1, 2, 3, 4, 5$. The delays on each of the links in ms are defined as follows:

$$\begin{aligned} ni \rightarrow c1 & : 40 \cdot i + 1; & pi \rightarrow c3 & : 40 \cdot i + 1 \\ mi \rightarrow c2 & : 40 \cdot i + 1; & qi \rightarrow c4 & : 40 \cdot i + 1 \end{aligned}$$

and the delays $c1 - c2$, $c2 - c3$, $c3 - c4$ are $10ms$. The capacities of all links are $10Mbps$. With this topology, the max-min fair shares are $0.5Mbps$ for 20 flows that uses link $c2 - c3$, and $1Mbps$ for other 10 flows ($n(i) \rightarrow m(i)$ and $p(i) \rightarrow q(i)$).

Each flow uses the standard TCP-SACK algorithm, with a packet size $1000B$. The aggressiveness of each flow is mainly determined by its RTT. The behavior of the network is evaluated with each link $c1 - c2$, $c2 - c3$ and $c3 - c4$ using: DropTail, RED, MLC(2), MLC(3), MLC(5), MLC(9), and MLC(17) with a queue size of 100 packets. MLC(a) parameters are: $l = a$, $\Delta_0 = 100ms$, $\gamma = 1.01$, $u_0 = 0.98$.

Normalized Jain's fairness index for vector that represents resource allocations $U = (U_1, \dots, U_N)$ in the network with max-min fair resource allocation given by vector $U_{mm} = (U_{1,mm}, \dots, U_{N,mm})$ is given by

$$j(U) = \frac{\left(\sum_{i=1}^N \frac{U_i}{U_{i,mm}}\right)^2}{N \sum_{i=1}^N \left(\frac{U_i}{U_{i,mm}}\right)^2}.$$

Its values for 7 schemes of interest are following:

$$j(U_{DTail}) = 0.345, j(U_{RED}) = 0.731, j(U_{MLC(2)}) = 0.846, j(U_{MLC(3)}) = 0.884,$$

$$j(U_{MLC(5)}) = 0.956, j(U_{MLC(9)}) = 0.989, j(U_{MLC(17)}) = 0.997.$$

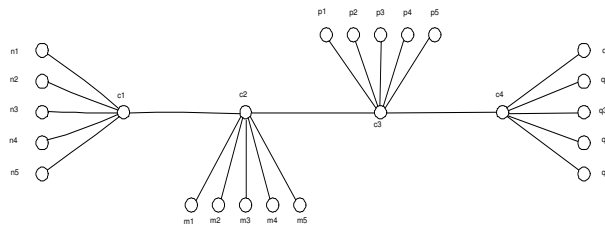


Fig. 2. Network topology

We can see significant unfairness in oblivious schemes: DropTail, RED. As we increase index of MLC scheme, we obtain share of bandwidth very close to max-min share as expected by Theorem 2.

4 Summary

In this paper we developed an AQM scheme for enforcing max-min fairness in TCP networks called MLC(l). MLC(l) is a stateless scheme and belongs to class of queue management schemes that we call power-drop. We showed analytically that by increasing index l , the resource allocation among TCP users using

network of $MLC(l)$ queues converge to max-min fair. The presented analytical findings are confirmed by packet level *ns2* simulations.

References

1. E. Altman, K. Avrachenkov, C. Barakat. "A stochastic model of TCP/IP with stationary random losses". *IEEE/ACM Transactions on Networking (TON)*, vol 13(2), 2005
2. A. Abouzeid, S. Roy. "Analytic understanding of RED gateways with multiple competing TCP flows". *Proc. of IEEE GLOBECOM*, 2000.
3. S. Athuraliya, D. Lapsley, S. Low. "Enhanced Random Early Marking algorithm for Internet flow control". *Proc. of IEEE INFOCOM*, Tel Aviv, Israel, 2000.
4. A. Das, D. Dutta, A. Goel, A. Helmy, J. Heidemann. "Low State Fairness: Lower Bounds and Practical Enforcement". *Proc. of the IEEE INFOCOM*, Miami, FL, USA, March 2005.
5. A. Demers, S. Keshav, S. Shenker. "Analysis and simulation of a fair queueing algorithm". *Proc. of ACM SIGCOMM*, Austin, TX, 1989.
6. V. Dumas, F. Guillemin, P. Robert. "A Markovian analysis of additive-increase multiplicative-decrease algorithms". *Adv. in Appl. Probab.* 34 (2002), no. 1, 85-111.
7. W. Feng, K.G. Shin, D.D. Kandlur, D. Saha. "The BLUE active queue management algorithms". *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, 513-528, August 2002.
8. S. Floyd. "HighSpeed TCP for Large Congestion Windows". RFC 3649, Experimental, December 2003.
9. S. Floyd. "Connections with Multiple Congested Gateways in Packet-Switched Networks Part 1: Oneway Traffic". *ACM Computer Communication Review*, 30 - 47, Vol. 21 , Issue 5, October 1991.
10. S. Floyd, V. Jacobson. "Random early detection gateways for congestion avoidance". *IEEE/ACM Transactions on Networking*, vol. 1, pp. 397 -413, Aug. 1993.
11. D. Katabi, M. Handley, C. Rohr. "Internet Congestion Control for Future High Bandwidth-Delay Product Environments". *Proc. of ACM SIGCOMM*, Pittsburgh, PA, USA, 2002.
12. T. Kelly. "Scalable TCP: Improving Performance in Highspeed Wide Area Networks". *ACM SIGCOMM Computer Communication Review* Volume 33, Issue 2, April 2003.
13. M. Kodialam, T. V. Lakshman, Shantidev Mohanty. "Runs based Traffic Estimator (RATE): A Simple, Memory Efficient Scheme for Per-Flow Rate Estimation". *Proc. of IEEE INFOCOM*, Hong Kong, March 2004.
14. T. V. Lakshman, 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, 336-350, June 1997.
15. D. Lin, R. Morris. "Dynamics of random early detection". *Proc. of ACM SIGCOMM*, Cannes, France, 1997.
16. S. Low, L. Andrew, B. Wyrowski. "Understanding XCP: Equilibrium and Fairness". *Proc. of IEEE Infocom*, Miami, FL, March 2005.
17. R. Mahajan, S. Floyd, D. Wetherall. "Controlling high-bandwidth flows at the congested router". *Proc. of IEEE ICNP*, Riverside, CA, USA, 2001.
18. J. Mo, J. Walrand. "Fair end-to-end window-based congestion control". *IEEE/ACM Transactions on Networking*, Vol. 8, No. 5 October, 2000.
19. T. J. Ott, T. V. Lakshman, L. H. Wong. "SRED: Stabilized RED". *Proc. IEEE INFOCOM*, New York, March 1999.
20. R. Pan, L. Breslau, B. Prabhakar, S. Shenker. "Approximate Fairness through Differential Dropping". *ACM SIGCOMM Computer Communication Review* Volume 33 , Issue 2, April 2003
21. R. Pan, B. Prabhakar, K. Psounis. "CHOKe: A stateless AQM scheme for approximating fair bandwidth allocation". *Proc. of IEEE INFOCOM*, Tel Aviv, Israel, 2000.
22. M. Shreedhar, G. Varghese. "Efficient fair queueing using deficit round-robin". *IEEE/ACM Transactions on Networking*, vol. 4, no. 3, June 1996.
23. R. Srikant. "The Mathematics of Internet Congestion Control". Birkhauser, 2004.
24. R. Stanojevic, R. Shorten. "AQM's for achieving fairness between competing TCP flows". Technical report, available online: www.hamilton.ie/person/rade/trm.pdf.
25. I. Stoica, S. Shenker, H. Zhang. "Core-Stateless Fair Queueing: A Scalable Architecture to Approximate Fair Bandwidth Allocations in High Speed Networks". *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, 33-46, February 2003.
26. A. Tang, J. Wang, S. H. Low. "Understanding CHOKe: throughput and spatial characteristics". *IEEE/ACM Transactions on Networking*. Vol 12 (4),2004.
27. B. Wyrowski, M. Zukerman. "MaxNet: A congestion control architecture for maxmin fairness". *IEEE Communications Letters*, vol. 6, no. 11, Nov. 2002, pp.512-514.
28. Y. Yang, S. Lam. "General AIMD Congestion Control". *Proc. ICNP 2000*, Osaka, Japan, 2000.