On the Isolation of Several Work-Conserving Scheduling Policies[1]

Antonio Fernández
Universidad Rey Juan Carlos
28933 Móstoles, Madrid, Spain
a.fernandez@escet.urjc.es

Abstract

In this paper we study the *isolation* of five work-conserving scheduling policies in connection-oriented packet-switched networks. We say that a policy has good isolation if its performance (end-to-end packet delay here) is not influenced by the session configuration. Here we study, by simulation on a very simple setup, how the average packet delay changes in one session when the length or number of the rest of sessions change (while the total rate at each link is preserved). In our study we consider two well-known scheduling policies, namely Weighted Fair Queueing (WFQ) and FIFO, a recently proposed label-based policy S-CEDF, and two more label-based policies we introduce here for connection-oriented networks. We observe that the performance of WFQ and FIFO tends to significantly vary under changing environments, while the label-based policies tend to be more stable. In particular, we observe that the end-to-end delay of WFQ and FIFO decreases when the length of the sessions competing with one given increases. When the environment changes by increasing the number of sessions, the delay under WFQ tends to decrease with the divisions, while the delay under FIFO tends to increase. Further work is needed to analyze these results, and to obtain empirical and analytical isolation bounds for a variety of policies.

I. INTRODUCTION

There is an increasing demand from the users for having some degree of Quality of Service (QoS) in the new packet-switched networks. This has generated a lot of effort to devise techniques, in order to provide such a QoS [10, 9]. One of the aspects to be controlled in a packet-switched network in order to guarantee QoS is the congestion at the switches. Congestion can increase the end-to-end delay of a packet and can even force the switch to drop packets. One of the means to reduce congestion is to use a "good" scheduling policies at the switches. For this reason, there is a lot of active work in offering "nice" scheduling algorithms and policies [4, 11, 14, 5, 17].

A. Scheduling Policies

One the most popular scheduling policies is the First-In-First-Out (FIFO) policy. It is simple and easy to implement. Unfortunately, it has been observed that FIFO is not stable (i.e. the congestion can grow unbounded) even in connection-oriented models of networks [1, 2] and, in general, cannot be used in order to guarantee QoS. This has forced to devise more complex scheduling policies.

Weighted Fair Queueing (WFQ) is one of the most popular alternatives to FIFO. Initially proposed by Demers et al. [8], its performance was analyzed by Parekh and Gallager [12, 13], obtaining an upper bound on the end-to-end delay experienced by any packet. This bound allows to know ahead the delay a session will experience. The bad news are that implementing WFQ is rather costly, which has driven research into finding alternative, easier to implement, scheduling algorithms with similar upper bounds on end-to-end delay [18, 17].

Under several simplifying assumptions, the upper bound on end-to-end delay derived for WFQ in [13] for a given session i is $O(K_i \times 1/\rho_i)$, where K_i is the number of links of the session path and ρ_i is the session rate. However, it has been shown that bounds of $O(K_i + 1/\rho_i)$ can be achieved with simple deadline-based randomized scheduling algorithms [3, 4]. In [4], Andrews and Zhang, proposed the Coordinated-Earliest-Deadline-First (CEDF) scheduling policy, for which they derived the analytical bound, and showed by simulation that the bound difference between WFQ and CEDF could be observed in simple setups. For the simulation they used a second simpler policy they called Simple-CEDF (S-CEDF). In this policy, instead of a deadline, a packet carries a label with similar function. This label is computed at packet arrival and is the sum of the arrival time and a random session-rate-dependent value. We say, hence, that S-CEDF is label-based.

In this paper we introduce two label-based scheduling policies to be used in connection-oriented networks. The first policy shall be called Longest-In-System (LIS) and was already studied in connectionless networks in [2]. It was shown there that it is universally stable (i.e. stable for all networks and all reasonable traffic patterns). As with S-CEDF, with LIS packets carry a label, which is the arrival time. Note that the label carries no information related to the session or the rate. The second policy, which we shall call Deterministic-CEDF (D-CEDF), tries to include in the label the session information of S-CEDF without its randomness, hence resulting in a deterministic policy. In D-CEDF the packet label is the sum of the arrival time plus a fixed session-rate-dependent value.

B. Isolation

Any good scheduling policy must satisfy the *isolation* property. This means that the end-to-end delay given to a session should not depend on the rest of the sessions. It is important to have policies that guarantee that traffic in one session does not degrade the delay in another. However, to our view, it is even more important that the delay on a session does not depend on the length or number of other sessions, as long as the accumulated rates at each link is kept constant. This is the aspect of isolation we study here.

[1]Partially supported by the Comunidad Autonónoma de Madrid.

0-7803-5794-9/99/$10.00 © 1999 IEEE

To our knowledge this is the first study on the isolation property of scheduling policies. This is surprising due to the amount of work devoted to popular policies like FIFO or WFQ.

C. Our Results

In this paper we evaluate by simulation the isolation properties of the above presented policies in a very simple network. In this network we change the length of the sessions and the number of them, and measure the delays observed in a session of reference, which is not changed. We are careful while changing the sessions that the total rate at each link has not changed.

From the results of the simulation we conclude that both WFQ and FIFO can have significant differences in delay depending on the parameters changed. In general, in both policies, the delay observed decreased when the session lengths was increased. However, when the change implied varying the number of sessions, in general FIFO's delay increased with the number while WFQ's decreased. In both kinds of changes the three label-based policies behave rather nicely, keeping their observed delay almost constant over the simulations.

The rest of the paper is structured as follows. In Section II we give the basic definitions and describe the model used. In Section III we present the simulation results. Finally, in Section IV we state the conclusions and future work.

II. Definitions

A. The Network Model

In this work we assume a packet-switched network formed by switches or routers (nodes, for short) connected by simplex links. The whole network can, hence, be modeled as a directed graph. Each packet traverses the network from its source node to its destination node by crossing the links in the appropriate direction.

We assume the switching works in a non-cut-through, non-preemptive fashion. This means that no packet can start to cross an output link of a node until it has been completely received (through an input link). Furthermore, once a packet has started to cross a link, it cannot be interrupted, and will continue until it is fully transmitted. We also assume that there is output buffering at the switches. This means that packets arriving at a node immediately appear at the queue of their output link from the node. Finally, we assume infinite buffer space at the nodes, so that packets are never dropped due to congestion.

For simplicity, we assume that all the packets have the same length and all the links have the same bandwidth. We also assume the switching, scheduling, and propagation times negligible. This allows us to study the evolution of the system as a synchronous process, in which there is a unit of time (a *step*) defined by the transmission time of a packet through a link. Then, we have a system in which, at each step, exactly one packet can cross each edge. For instance, if we have an ATM network with links running at 40 Mbps, an ATM cell takes 10.6 microseconds to cross a link. In our model, 10.6 microseconds would be the step duration.

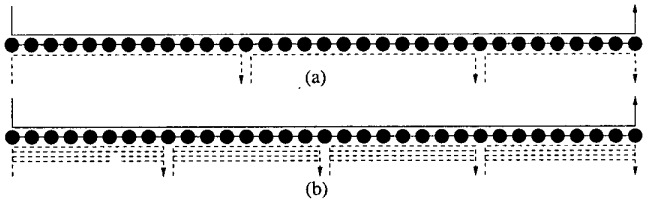

Figure 1: Network and sessions used in our simulations. The session of reference is drawn with a solid line and the competing sessions are drawn with dashed lines.

The *end-to-end delay* of a packet is the number of steps it takes the packet to reach its destination node, measured from the time it arrived to its source node. Under our assumptions, it should be clear that this time is the distance from its source to its destination node plus the steps the packet has been waiting at some node due to scheduling decisions.

B. The Session Model

We consider here connection-oriented networks, i.e. networks in which all the packet traffic is grouped in connections or *sessions*. Each session has a source and a destination node, which are the corresponding source and destination nodes of each packet belonging to the session.

The flow of new packets arriving to a session i is described by a pair (σ_i, ρ_i), as introduced by Cruz [6, 7], where $0 \leq \rho_i \leq 1$ is the rate and $\sigma_i \geq 1$ is the burst size of the session. If $A_i(t_1, t_2)$ denotes the number of packets arrived to the network belonging to session i in the time interval $(t_i, t_2]$, then $A_i(t_1, t_2) \leq \sigma_i + \rho_i(t_2 - t_1)$. From this, it can be simply observed that the long-term arrival rate is ρ_i, but bursts of up to σ_i packets are allowed. We say, then, that the session arrivals are *leaky-bucket constrained* with bucket size σ_i. As usual, we consider that no link is traversed by a collection of sessions such that their cumulative rate is more than 1.

C. Our Network

In our experiments we shall use a network consisting of a line of 32 links. In this network there is a session crossing all the links, which will be the *session of reference* r. Competing with packets of session r for the use of the links there will be packets belonging to a number of other sessions, which we will collectively name *competing sessions*. Similar setups have been used in previous simulation studies [15, 4, 16]. In all the sets of experiments we keep constant the pair (σ_r, ρ_r) of session r and the sum of the rates in each edge. We study the change of the end-to-end delay of session-r packets when the length or the number of competing sessions varies.

In our first collection of experiments, session r is sharing each link with exactly one competing session, and all the competing sessions (except maybe the last one) have the same length. In the resulting network the first node is source for one competing session, the last node is destination for one competing session, and except those, each destination of a

competing session is source for another (the next) competing session. Figure 1.(a) shows one of the networks studied in which two competing sessions have length 12 and the last one has length 8. In this collection of experiments we try to observe the changes in the end-to-end delay of session-r packets induced by the different lengths of the competing sessions.

In the second collection, we fix the length of the competing sessions to 8 links. Session r still shares each edge with the same number of competing sessions, but we change how many of them to observe the influence of this change in the end-to-end delay of session-r packets. Figure 1.(b) shows one of the networks studied, in which r shares each edge with 4 competing sessions.

D. Scheduling Policies

FIFO. FIFO chooses as next packet to cross a link the packet that has been waiting the longest at the node for the link. It is very simple to implement but, as we said, is not always stable.

Weighted Fair Queueing. WFQ attempts to emulate Generalized Processor Sharing (GPS), a scheduling policy for the fluid model, in which packets from all sessions waiting for a link cross it simultaneously. WFQ is a "discretization" of GPS, in which the first packet that would finish crossing the link under GPS is given priority. In general, GPS can use weights to give priority to one session over the other, and determine the portion of the link allocate to a session based on those weights. Here we assume that the weight of a session i is its rate ρ_i, and that the portion of the link e assigned to session i is $\rho_i / \sum_{j \in B_e} \rho_j$, where B_e is the set of sessions with packets waiting to cross e. Using these weights we obtain a special case of WFQ known as Rate Proportional Processor Sharing.

Longest-in-System. LIS gives priority to the packet that has been the longest in the network. As we said, LIS is universally stable, but does not take into account the sessions rates. A simple implementation of LIS attaches to each packet its arrival time as a label, and chooses at each link the packet with smallest label as next packet to cross the link.

Simple-Coordinated-Earliest-Deadline-First. S-CEDF assigns to each session-i packet a label which is its arrival time plus a value chosen uniformly at random from the interval $[0, 1/\rho_i]$. This label is incremented each time the packet crosses a link. The packet with the smallest label is given priority. As we mentioned, S-CEDF presents nice simulation performance. However, the analytical bounds obtained for its more complicated version CEDF have a probabilistic component.

Deterministic-Coordinated-Earliest-Deadline-First. D-CEDF assigns to each session-i packet a label which is its arrival time plus $1/\rho_i$, and which is incremented each time the packet crosses a link. As the two previous policies, the packet with the smallest label is given priority. The idea behind this policy is that it is basically S-CEDF with the randomness removed. It takes into account the session rates and is simple to implement.

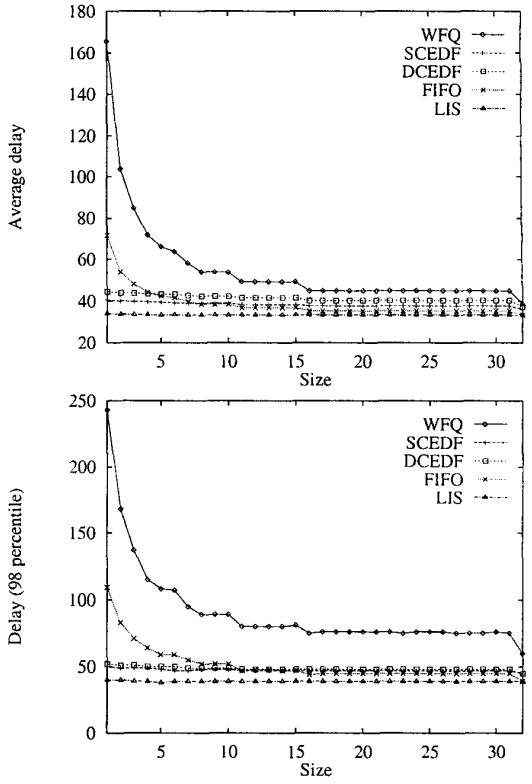

Figure 2: Average and 98% percentile of the session-r delays when the length of the competing session is varied, for $\sigma_i = 1$.

III. Simulation Results

In this section we present the results obtained from the simulation experiments. Unless otherwise stated, in all the experiments the total link rate used was 0.8, with $\rho_r = 0.1$. That left 0.7 for the cumulative rate of the competing sessions in each link. We have also done experiments in which the total link rate is 0.9, but we have not observed a significant difference with the results presented here. Most experiments have been done with two kinds of burstiness. In the first, $\sigma_i = 1$ for each session i, i.e. there is no burstiness. In the second, we allow some burstiness by setting $\sigma_i = 10$, for each session i.

A. Varying the Length of the Competing Sessions

Figure 2 shows the delays observed when the length of the competing sessions is varied from 1 to 32, without burstiness. As it can be seen, WFQ is the policy with largest delays, and less uniform behavior. The average delay for the length-1 case is around three times that of a case with length above 20, and the same can be said about the 98% percentile of the delay. FIFO has also different delays for different lengths. The three label-based policies have almost the same delay for any session length. From them, LIS is the one with smallest delays, while D-CEDF has slightly larger delays than S-CEDF.

Figure 3 shows the average delays resulting from a similar experiment but with some burstiness. As a consequence, the absolute values on the delays are substantially higher, but the

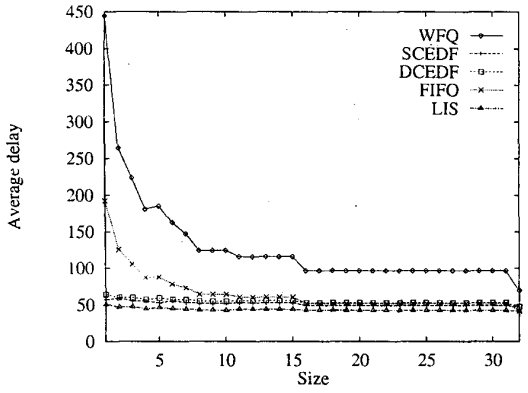

Figure 3: Average session-r delays when the length of the competing session is varied, for $\sigma_i = 10$.

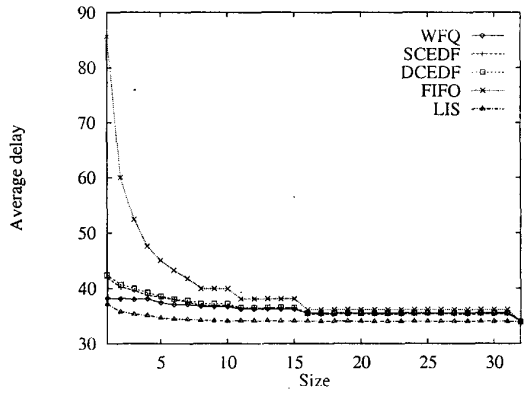

Figure 4: Average session-r delays when the length of the competing session is varied, for $\sigma_i = 1$ and $\rho_r = 0.4$.

general behavior is the same.

It is worth noting that the lack of insulation of WFQ is influenced by the relative rates of the sessions. In Figure 4 we have the delays observed when the rate of all the sessions is 0.4. In this case, the WFQ curve is rather flat, showing a higher level of insulation. FIFO's behavior, on the contrary, does not show a big change.

B. *Varying the Number of Competing Sessions*

In Figure 5 we show the delays observed when the number of parallel competing sessions is varied from 1 to 9. It can be observed that again WFQ and FIFO have larger variations than the label-based policies. However, in this case the delays under WFQ decrease (except from 1 to 2) when the number increases, while the delays under FIFO increase with the number. In both cases, the differences are not as drastic as in the previous section. Here, the label-based policies does not show such a clear insulation as in the previous section. As before, LIS is the policy that presents the smallest delays, but they increase with the number of parallel sessions.

It is interesting to observe that when the number of parallel sessions is varied the degree of burstiness seems to be of more importance than in the previous section. In Figure 6 we present

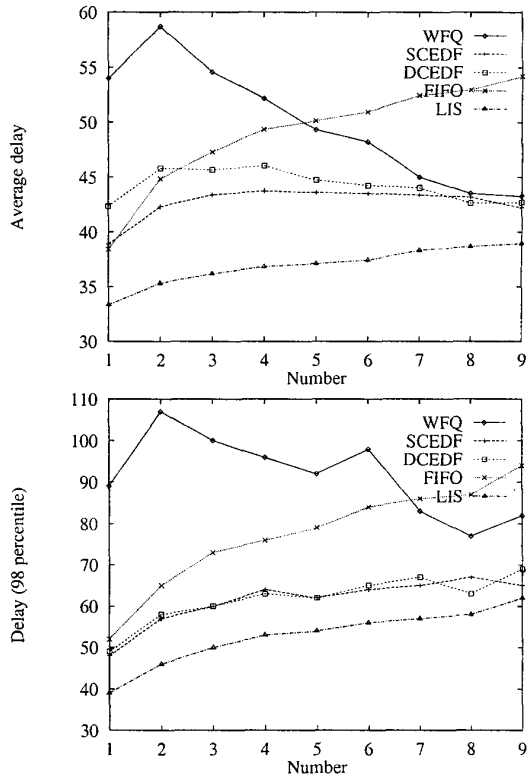

Figure 5: Average and 98% percentile of the session-r delays when the number of competing session is varied, for $\sigma_i = 1$.

the delays with burstiness $\sigma_i = 10$, and it can be observed that the degree of variation with the number is higher for all the policies.

Finally, in Figure 7 we present the delays experienced by the competing sessions. They increase with the number of parallel sessions for all the policies, and at about the same degree.

IV. Conclusions and Future Work

This paper attempts to be one of the first works on the isolation property of scheduling policies. In this paper we have evaluated by simulation the variation of the end-to-end delay under various scheduling policies when the length or the number of the sessions of the environment change. We have contrasted the results for five work-conserving policies.

There are many lines of future work. First, in order to contrast the results presented, similar experiments should be performed on a number of other setups. Second, we have observed that the label-based policies outperform, in our setup, WFQ and FIFO. However, we have not been able to identify which property makes them work better. We plan to attempt to derive analytical bounds under these polices. In any case, much more empirical and analytical work is needed to study the isolation of these and many more scheduling policies.

Acknowledgments

We would like to thank Matthew Andrews and Lisa Zhang for many useful discussions and for their simulation code.

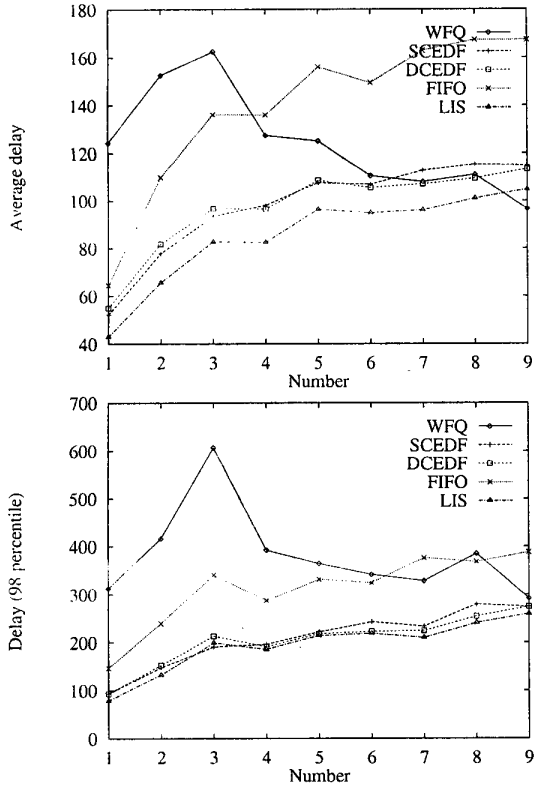

Figure 6: Average and 98% percentile of the session-r delays when the number of competing session is varied, for $\sigma_i = 10$.

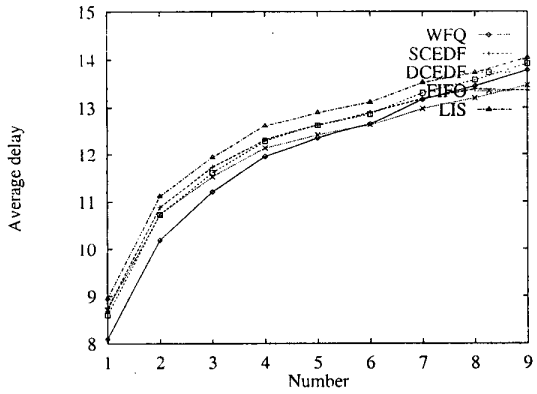

Figure 7: Average delays of the competing sessions when their number is varied, for $\sigma_i = 1$.

V. References

[1] M. Andrews, "Instability of FIFO in networks with bounded session rates," Unpublished manuscript, 1999.

[2] M. Andrews, B. Awerbuch, A. Fernández, J. Kleinberg, T. Leighton, and Z. Liu, "Universal stability results for greedy contention-resolution protocols," in *Proc. of the 37th Symp. on Found. of Comp. Sc.*, (Burlington, VT), pp. 380–389, Oct. 1996.

[3] M. Andrews, A. Fernández, M. Harchol-Balter, T. Leighton, and L. Zhang, "General dynamic routing with per-packet delay guarantees of O(distance + 1/session rate)," in *Proc. of the 38th Symp. on Found. of Comp. Sc.*, (Miami Beach, FL), pp. 294–302, Oct. 1997.

[4] M. Andrews and L. Zhang, "Minimizing end-to-end delay in high-speed networks with a simple coordinated schedule," in *Proc. of the Conf. on Comp. Comm., INFOCOM'99*, Mar. 1999.

[5] J. A. Cobb, M. G. Gouda, and A. El-Nahas, "Time-shift scheduling — fair scheduling of flows in high-speed networks," *IEEE/ACM Trans. on Networking*, vol. 6, no. 3, pp. 274–285, June 1998.

[6] R. L. Cruz, "A calculus for network delay, part I: Network elements in isolation," *IEEE Trans. on Information Theory*, vol. 37, no. 1, pp. 114–131, Jan. 1991.

[7] R. L. Cruz, "A calculus for network delay, part II: Network analysis," *IEEE Trans. on Information Theory*, vol. 37, no. 1, pp. 132–141, Jan. 1991.

[8] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *Internetworking: Research and Experience*, vol. 1, no. 1, pp. 3–26, 1990.

[9] P. Ferguson and G. Huston, *Quality of Service: Delivering QoS on the Internet and in Corporate Networks*. New York: John Wiley & Sons, Inc., 1998.

[10] S. Keshav, *An Engineering Approach to Computer Networking: ATM Networks, the Internet, and the Telephone Network*. Addison-Wesley, 1997.

[11] S.-K. Kweon and K. G. Shin, "Providing deterministic delay guarantees in ATM networks," *IEEE/ACM Trans. on Networking*, vol. 6, no. 6, pp. 838–850, Dec. 1998.

[12] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: The single node case," *IEEE/ACM Trans. on Networking*, vol. 1, pp. 344–357, June 1993.

[13] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: The multiple node case," *IEEE/ACM Trans. on Networking*, vol. 2, pp. 137–150, Apr. 1994.

[14] D. Saha, S. Mukherjee, and S. K. Tripathi, "Carry-over round robin: A simple cell scheduling mechanism for ATM networks," *IEEE/ACM Trans. on Networking*, vol. 6, no. 6, pp. 779–796, Dec. 1998.

[15] H. Schulzrinne, J. Kurose, and D. Towsley, "An evaluation of scheduling mechanisms for providing best-effort, real-time communication in wide-area networks," in *Proc. of the Conference on Computer Communications, INFOCOM'94*, (Toronto, Canada), June 1999.

[16] D. Stiliadis, *Traffic Scheduling in Packet-Switched Networks: Analysis, Design, and Implementation*. PhD thesis, U. of California Santa Cruz, June 1996.

[17] D. Stiliadis and A. Varma, "Efficient fair queueing algorithms for packet-switched networks," *IEEE/ACM Trans. on Networking*, vol. 6, pp. 175–185, 1998.

[18] D. Stiliadis and A. Varma, "Rate-proportional servers: A design methodology for fair queueing algorithms," *IEEE/ACM Trans. on Networking*, vol. 6, pp. 164–174, 1998.