Dedicated Channels as an Optimal Network Support for Effective Transfer of Massive Data

Sergey Gorinsky
Applied Research Laboratory, CSE Department
Washington University in St. Louis
One Brookings Drive, Campus Box 1045
St. Louis, MO 63130-4899, USA
gorinsky@wustl.edu

Nageswara S. V. Rao
Computer Science and Mathematics Division
Oak Ridge National Laboratory
One Bethel Valley Road, P.O. Box 2008
Oak Ridge, TN 37831-6016, USA
raons@ornl.gov

***Abstract*— Instantaneous Fair Sharing (IFS) is a traditional network ideal prescribing to share the network capacity among competing applications fairly during any infinitesimal time interval. In this paper, we argue that IFS is an inappropriate ideal for the application of massive data transfers where the primary goal is to minimize message transfer times. We propose an alternative paradigm of Virtual Finish Time First (ViFi) scheduling that dedicates the entire capacity to one message at a time in the order of message finish times under IFS. Unlike Shortest Remaining Time First and other earlier algorithms for dedicated scheduling, ViFi provides a remarkable guarantee of delivering each message no later than under IFS. Our analysis and simulations show the dedicated ViFi scheduling offers significant reductions in the average transfer time. The above properties make ViFi a promising approach for resource allocation in emerging dedicated-channel networks that enable advance reservation of end-to-end channels between hosts.**

I. Introduction

While a typical ideal for Internet fairness is Instantaneous Fair Sharing (IFS) that prescribes to share a contended capacity among applications instantaneously, this paper argues for relaxing the IFS ideal and assessing fairness at the meaningful for applications granularity of messages. We propose an alternative Virtual Finish Time First (ViFi) paradigm and evaluate its suitability for massive data transfers over dedicated-channel networks, which are described in Section II. Sections III and IV discuss IFS and ViFi. Sections V and VI present comparative analysis and simulations of these two paradigms. Finally, Section VII sums up our contributions.

II. Internet and Dedicated-Channel Networks

The tremendous success of the Internet is partly due to its architectural minimalism which enables easy formation of best-effort networks from diverse communication devices and media. The lightness of the Internet architecture comes with suboptimality of services offered to applications: since network-layer and transport-layer protocols in the Internet are oblivious of applications, the protocols pursue generic goals that are imperfectly aligned with specific needs of different applications. In particular, the Internet is an inadequate platform for instrument control and other applications that require stringent delay and capacity assurances.

The suboptimality of the Internet support for large-scale scientific applications has lead to emergence of UltraScience Net (USN) [13], Circuit-switched High-speed End-to-End Transport ArcHitecture (CHEETAH) [17], and other networks that accept reservations for end-to-end channels along high-speed network links. This mechanism of dedicated channels enables an application to utilize the allocated network resources without interference from cross traffic. In comparison to the Internet, dedicated-channel networks are clearly more preferable for applications that require assured network services [15]. Moreover, best-effort applications also benefit from dedicated capacity allocation because the latter improves awareness of the applications about the available capacity and thereby promotes a more efficient utilization of network resources [16]. In this paper, we reach a surprising conclusion that dedicated capacity allocation remains preferable even after fairness considerations are taken into account.

III. Ideal of Instantaneous Fair Sharing

A traditional ideal in designing a computer network is to share the network capacity among competing applications fairly during any infinitesimal time interval. Our paper refers to this ideal as *Instantaneous Fair Sharing* (*IFS*). Prominent examples of IFS include maxmin fairness [2], its weighted version [2], and proportional fairness [9]. When a network link is the only bottleneck resource, a widely accepted form of IFS is Generalized Processor Sharing (GPS) which defines a fair instantaneous rate for application i at time t as

$$r_i(t) = \left(\frac{w_i}{\sum_{i=1}^{n} w_i} \right) c \qquad (1)$$

where n is the number of applications contending for link capacity c at time t, and w_i is a positive weight associated with application i [11].

The IFS ideal is unattainable in packet-switching networks because the latter allocate the capacity of a link to one packet at a time. A lot of research has been conducted on approximating IFS with router-assisted or purely end-to-end designs.

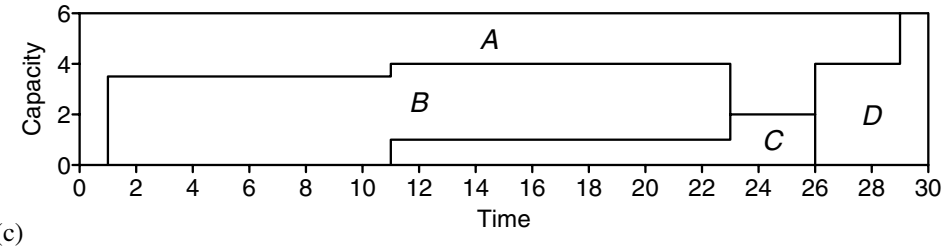
(a)

Message	Size	Weight	Arrival time
A	72	2	0
B	72	3	1
C	18	1	11
D	18	4	26

(b)

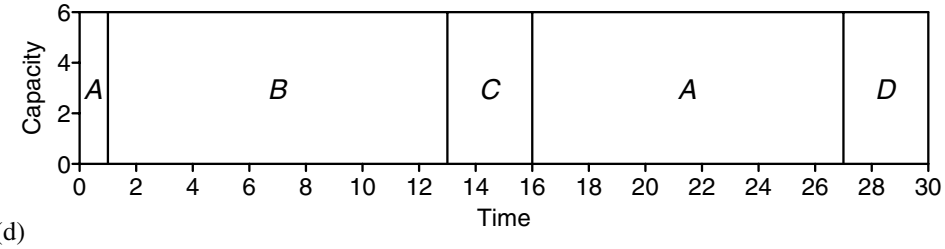
Time		0	1	11	23	26	29	30
Finish	for message A	12	24	27	27	29	**29**	-
time	for message B	-	21	23	**23**	-	-	-
under	for message C	-	-	26	26	**26**	-	-
GPS	for message D	-	-	-	-	30	30	**30**

(c)



Message	Transfer time
A	29
B	22
C	15
D	4

(d)



Message	Transfer time
A	27
B	12
C	5
D	4

Fig. 1. Example with four messages contending for capacity of 6: (a) characteristics of the messages, (b) dynamics of finish times under GPS, (c) capacity allocation under GPS, and (d) capacity allocation under ViFi.

Packet-by-packet GPS (PGPS) [11], also known as Weighted Fair Queuing (WFQ) [4], Deficit Round Robin (DRR) [14], Start-time Fair Queueing (STFQ) [7], Transmission Control Protocol (TCP) [8], and Rate Control Protocol (RCP) [5] are salient examples of such designs.

IV. ViFi Scheduling

In this section, we argue that IFS is an inappropriate fairness ideal for massive data transfers, a prominent type of large-scale scientific applications in dedicated-channel networks. For a data transfer, the primary objective is to minimize the message transfer time, i.e., the amount of time to deliver the entire message from the sending end to receiving end of the application. Although Shortest Remaining Time First (SRTF) and other disciplines that dedicate the whole capacity to one message at a time are known for reducing the average message transfer time considerably [1], [10], two factors have impeded adoption of dedicated scheduling for network resource allocation. The first is a requirement of knowing message sizes in advance; this factor is of little import for dedicated-channel networks where capacity reservation requests are already based on a priori estimates of message sizes. The second is a potential for starvation of a long message by a stream of shorter messages. To alleviate the unfairness concern, we introduce a concept of Virtual Finish Time First (ViFi) scheduling and define a ViFi counterpart for any IFS discipline as follows:

Definition 1: Virtual Finish Time First (ViFi) schedules messages preemptively in the order of their finish times under a corresponding IFS discipline.

A remarkable property of ViFi is the guarantee that no message finishes later than under IFS:

Theorem 1: Transfer time for any message under ViFi is smaller or equal than under the respective IFS.

Proof: Let $i = 1, \ldots, n$ enumerate all messages in an IFS schedule in the order of their finish times. For each i from 1 to $n-1$, and for each message j that shares capacity with i, consider time interval $[a_i,\ f_i]$, where a_i and f_i are the arrival and finish times of i in the current schedule, and reallocate capacity consumed by j from the beginning of $[a_i,\ f_i]$ to i in exchange for capacity consumed by i at the end of $[a_i,\ f_i]$ until the capacity allocations for i and j do not overlap during any time interval; the reallocation reduces the finish time of i and preserves the finish time of j. The resulting message-grained schedule serves messages preemptively in the order of their IFS finish times. Thus, this is a ViFi schedule. By construction, the ViFi schedule provides each message with a smaller or equal transfer time than under the IFS discipline. ■

Throughout the rest of the paper, we study ViFi scheduling in contrast with the GPS instantiation of IFS. Figure 1 illustrates advantages of ViFi over GPS in a simple example. Figure 1a presents sizes, weights, and arrival times for four messages A, B, C, and D. Figure 1b shows how arrivals of messages affect GPS finish times of messages that have arrived earlier. Figures 1c and 1d depict the capacity allocation under GPS and ViFi respectively. By scheduling messages in the order of their GPS finish times, ViFi preserves the transfer time for message D, slightly improves the transfer time for message A, cuts the the transfer time for message B about

in half, and reduces the transfer time for message C by the factor of three.

V. Analysis

In this section, we analyze the performance benefits of ViFi scheduling in a static scenario where n messages with sizes s_i and weights w_i (where $i = 1, \ldots, n$ and $\frac{s_k}{w_k} \leq \frac{s_j}{w_j}$ if $k < j$) arrive simultaneously and contend for capacity c. Under GPS, the k-th message attains transfer time

$$t_k^{\text{GPS}} = \frac{\sum_{i=1}^{k} s_i}{c} + \frac{\sum_{i=k+1}^{n} w_i}{w_k} \cdot \frac{s_k}{c}. \tag{2}$$

Under ViFi, the transfer time of the k-th message becomes

$$t_k^{\text{ViFi}} = \frac{\sum_{i=1}^{k} s_i}{c}. \tag{3}$$

Note that $t_k^{\text{ViFi}} < t_k^{\text{GPS}}$ for $1 \leq k \leq n-1$ and $t_n^{\text{ViFi}} = t_n^{\text{GPS}}$. Hence, dedicating the entire capacity to one message at a time improves the best transfer time from

$$t_{\text{min}}^{\text{GPS}} = \frac{\sum_{i=1}^{n} w_i}{w_1} \cdot \frac{s_1}{c} \quad \text{to} \quad t_{\text{min}}^{\text{ViFi}} = \frac{s_1}{c}, \tag{4}$$

which is by the factor of n when all the weights are equal. The dedicated allocation preserves the worst transfer time

$$t_{\text{max}}^{\text{GPS}} = t_{\text{max}}^{\text{ViFi}} = \frac{\sum_{i=1}^{n} s_i}{c} \tag{5}$$

and cuts the average transfer time from

$$t_{\text{ave}}^{\text{GPS}} = \frac{\sum_{k=1}^{n} (n-k+1) s_k}{nc} + \frac{\sum_{k=1}^{n-1} \left(\frac{s_k}{w_k} \sum_{i=k+1}^{n} w_i \right)}{nc} \tag{6}$$

to

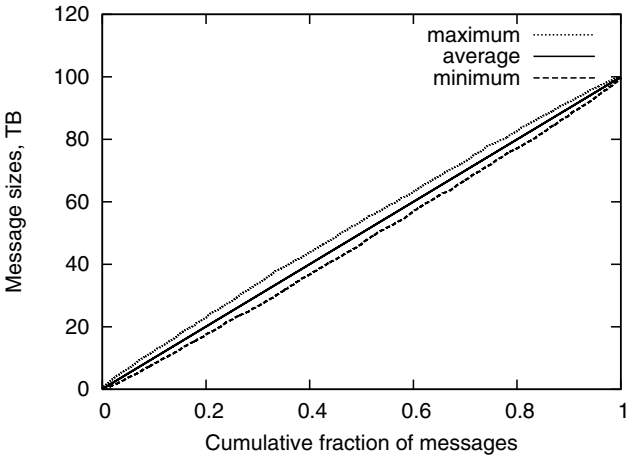
$$t_{\text{ave}}^{\text{ViFi}} = \frac{\sum_{k=1}^{n} (n-k+1) s_k}{nc} \tag{7}$$

which is about in half for equal weights and large n. Although the twofold average improvement is not substantial for small messages, petabyte data transfers over Tbps networks would benefit significantly from halving the average transfer time, e.g., from two hours to one.

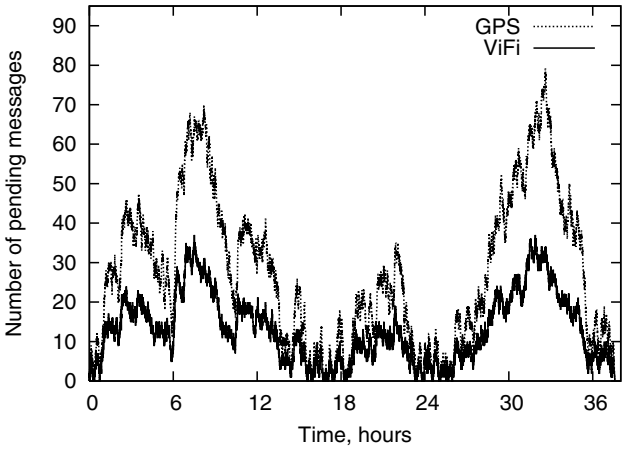
VI. Simulations

We continue our comparative evaluation of GPS and ViFi by simulating scenarios where messages arrive at different times. The code and running instructions for all the reported simulations are available at our web site [6].

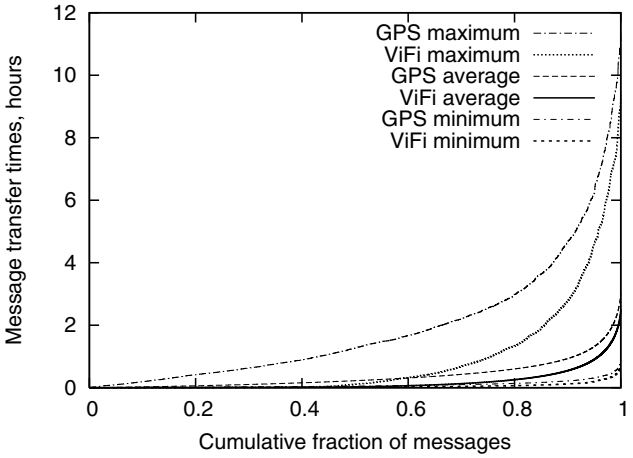
First, we experiment in settings where 3000 messages arrive according to a Poisson process with the average rate of one message per 45 seconds, message weights are uniformly drawn from set {1, 2, 3, 4}, and message sizes are distributed uniformly between 100 GB and 100 TB. The messages compete for capacity of 10 Tbps. To make our results



(a) Summary of the message size distributions in the 10000 experiments



(b) Number of pending messages during the first of the experiments



(c) Summary of the message transfer times in the 10000 experiments

Fig. 2. Simulations with uniformly distributed message sizes: capacity of 10 Tbps, 3000 messages, Poisson arrivals, average arrival rate of one message per 45 seconds, uniformly distributed message weights from set {1, 2, 3, 4}, message sizes between 100 GB and 100 TB.

statistically significant, we repeat the simulation 10000 times. Figure 2a summarizes the message size distributions in these 10000 experiments. Figure 2b tracks the number of pending messages during the first of the experiments. In this particular simulation, the average number of pending messages reduces from 27.44 with GPS to 12.94 with ViFi. The reduction is typical and consistently results in lower transfer times under ViFi. Figure 2c reflects this positive impact of ViFi on message transfer times in the 10000 experiments. We also compute the average transfer time as an average over both the number of messages and the number of experiments. In comparison to GPS, ViFi decreases the average number of pending messages from 30.35 to 14.25 and the average transfer time from 22.86 minutes to 10.73 minutes.

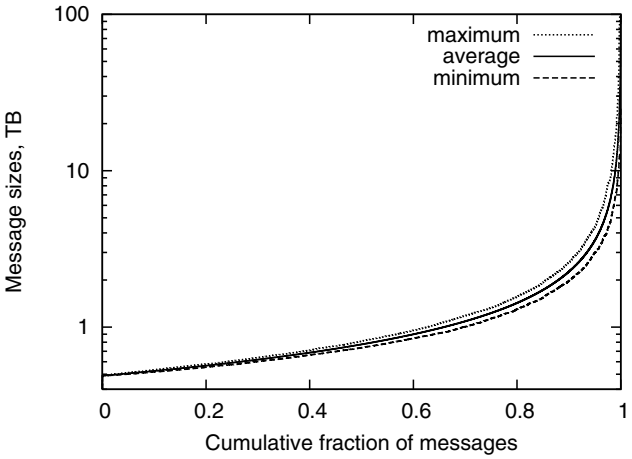
Although distributions of message sizes in future large-scale scientific applications are unknown, it has been argued that Internet applications generate messages with sizes conforming to heavy-tailed distributions [3], [12]. Hence, we repeat the above experiments for message sizes from the Pareto distribution with Pareto index 1.5 and minimum size 500 GB. The messages arrive with the average rate of one message per second. The other experimental settings remain the same. Figure 3a presents a summary of the message size distributions in the 10000 simulation runs. As Figure 3b shows for one experiment, ViFi manages to keep the number of pending messages at substantially lower levels than GPS. Over all 10000 experiments, the average number of pending messages decreases from 144.2 with GPS to 29.7 with ViFi. The decrease is accompanied by a similar fivefold reduction in the average transfer time from 3.07 minutes with GPS to 37.89 seconds with ViFi. Figure 3c plots the message transfer times under GPS and ViFi in more detail.

We conduct the 10000-run experimental series for different arrival rates of Pareto-sized messages. Figure 4a demonstrates that the average transfer time under ViFi is consistently lower than under GPS. We attribute the improvement to the ability of ViFi to reduce the number of pending messages. Figure 4b confirms this assertion by showing for both GPS and ViFi that the average transfer time and average number of pending messages are perfectly correlated.

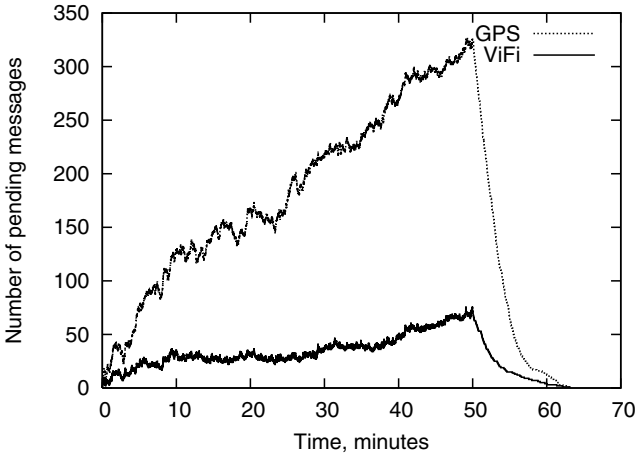
In the last series of our experiments, weights of Pareto-sized messages are uniformly drawn from the set of all integers between 1 and $1 + d$. We refer to the integer parameter d as a *weight diversity* and repeat the simulation 10000 times for each value of d between 0 and 100. Figure 5 shows that the average transfer time under either GPS or ViFi increases as the weight diversity grows: under both disciplines, the longest 20% of the messages – which contribute the most to the average transfer time – finish later with a larger weight diversity.

VII. Conclusion

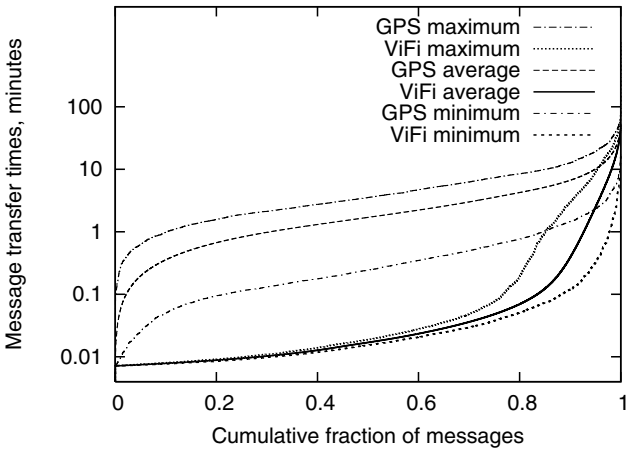
In this paper, we demonstrated that the traditional network ideal of Instantaneous Fair Sharing is imperfectly aligned with the needs of massive data transfers. We proposed an alternative paradigm of ViFi scheduling that dedicates the entire contended capacity to one message at a time in the order



(a) Summary of the message size distributions in the 10000 experiments

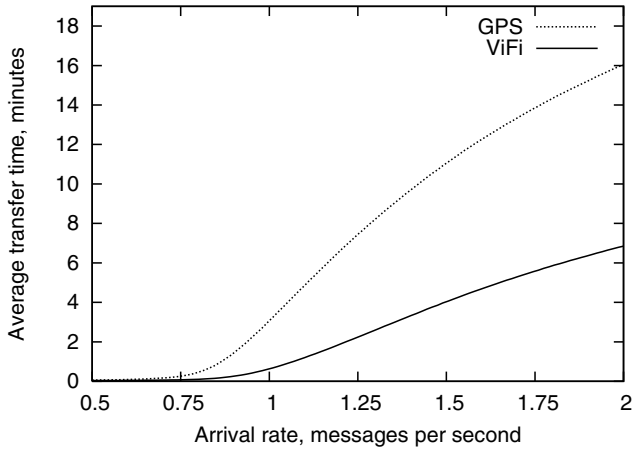


(b) Number of pending messages during the first of the experiments

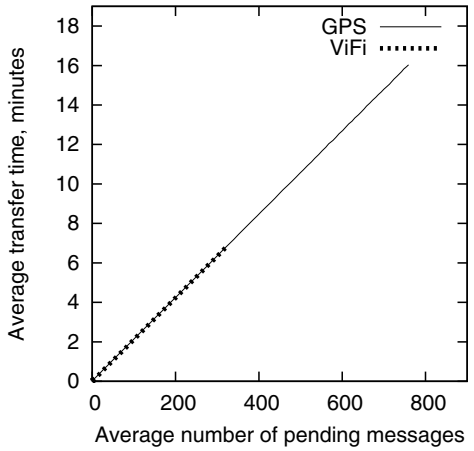


(c) Summary of the message transfer times in the 10000 experiments

Fig. 3. Simulations with Pareto-sized messages: capacity of 10 Tbps, 3000 messages, Poisson arrivals, average arrival rate of one message per second, uniformly distributed message weights from set $\{1, 2, 3, 4\}$, Pareto-distributed message sizes with Pareto index 1.5 and minimum size 500 GB.



(a) Consistent advantage of ViFi over GPS



(b) Perfect correlation between the average transfer time and average number of pending messages

Fig. 4. Dependence of the average transfer time on the arrival rate: averaging over 10000 experiments, capacity of 10 Tbps, 3000 messages, Poisson arrivals, uniformly distributed message weights from set $\{1, 2, 3, 4\}$, Pareto-distributed message sizes with Pareto index 1.5 and minimum size 500 GB.

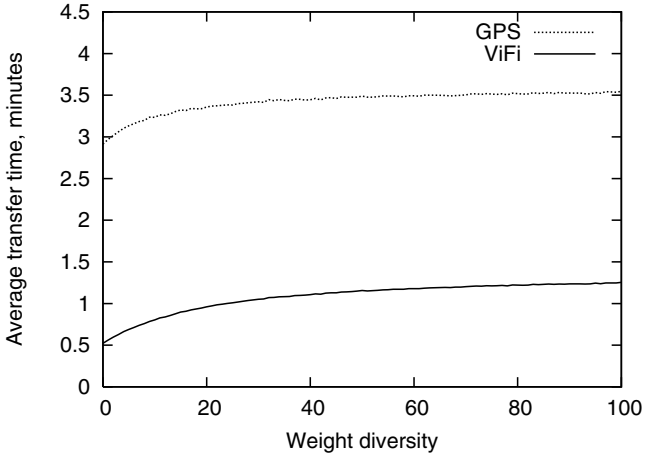


Fig. 5. Impact of the weight diversity on the average transfer time: averaging over 10000 experiments, capacity of 10 Tbps, 3000 messages, Poisson arrivals, average arrival rate of one message per second, Pareto-distributed message sizes with Pareto index 1.5 and minimum size 500 GB.

of message finish times under IFS. ViFi guarantees a smaller or equal transfer time for any message and significantly decreases the average transfer time. We plan to extend the presented evaluation of ViFi to include probabilistic analysis and more comprehensive experiments. We also intend to use ViFi as a basis for a practicable USN scheduler.

ACKNOWLEDGMENTS

This research was made possible at Oak Ridge National Laboratory by funds provided by the U.S. DOE (contract DE-AC05-00OR22725), DARPA (MIPR K153), and NSF (grants ANI-0229969 and ANI-0335185). The author at Washington University in St. Louis is thankful to Christoph Jechlitschek, Maxim Podlesny, and Manfred Georg for their assistance and discussions.

REFERENCES

[1] N. Bansal and M. Harchol-Balter. Analysis of SRPT Scheduling: Investigating Unfairness. In *Proceedings ACM SIGMETRICS 2001*, June 2001.

[2] D. Bertsekas and R. Gallager. *Data Networks*. Prentice-Hall, 1987.

[3] M.E. Crovella and A. Bestavros. Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. *IEEE/ACM Transactions on Networking*, 5(6):835–846, December 1997.

[4] A. Demers, S. Keshav, and S. Shenker. Analysis and Simulation of a Fair Queueing Algorithm. In *Proceedings ACM SIGCOMM 1989*, September 1989.

[5] N. Dukkipati, M. Kobayashi, R. Zhang-Shen, and N. McKeown. Processor Sharing Flows in the Internet. In *Proceedings International Workshop on Quality of Service (IWQoS 2005)*, June 2005.

[6] S. Gorinsky. Simulation Suite for Comparative Studies of ViFi and GPS. www.arl.wustl.edu/~gorinsky/ViFi, May 2006.

[7] P. Goyal, H.M. Vin, and H. Cheng. Start-time Fair Queueing: A Scheduling Algorithm for Integrated Services Packet Switching Networks. In *Proceedings ACM SIGCOMM 1996*, August 1996.

[8] V. Jacobson. Congestion Avoidance and Control. In *Proceedings ACM SIGCOMM 1988*, August 1988.

[9] F. Kelly. Charging and Rate Control for Elastic Traffic. *European Transactions on Telecommunications*, 8(1):33–37, January 1997.

[10] A. Kherani and R. Nunez-Queija. TCP as an Implementation of Age-Based Scheduling: Fairness and Performance. In *Proceedings IEEE INFOCOM 2006*, April 2006.

[11] 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 Transactions on Networking*, 1(3):344–357, June 1993.

[12] V. Paxson and S. Floyd. Wide-Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, June 1995.

[13] N.S.V. Rao, W.R. Wing, S.M. Carter, and Q. Wu. UltraScience Net: Network Testbed for Large-Scale Science Applications. *IEEE Communications*, 43(11):S12–S17, November 2005.

[14] M. Shreedhar and G. Varghese. Efficient Fair Queueing Using Deficit Round Robin. In *Proceedings ACM SIGCOMM 1995*, September 1995.

[15] Terascale Supernova Initiative. http://www.phy.ornl.gov/tsi/.

[16] Q. Wu and N.S.V. Rao. A Class of Reliable UDP-Based Transport Protocols Based on Stochastic Approximation. In *Proceedings IEEE INFOCOM 2005*, March 2005.

[17] X. Zheng, M. Veeraraghavan, N.S.V. Rao, Q. Wu, and M. Zhu. CHEETAH: Circuit-Switched High-Speed End-to-End Transport Architecture Testbed. *IEEE Communications*, 43(8):S11–S17, August 2005.