City Research Online

City, University of London Institutional Repository

Permanent repository link: http://openaccess.city.ac.uk/id/eprint/22827/

Link to published version: http://dx.doi.org/10.1109/ICUMT.2012.6459727

City Research Online: http://openaccess.city.ac.uk/   publications@city.ac.uk

Multi-Channel Deficit Round-Robin Scheduling for Hybrid TDM/WDM Optical Networks

Mithileysh Sathiyanarayanan and Kyeong Soo Kim

College of Engineering, Swansea University

Swansea, SA2 8PP, United Kingdom

Email:{m.sathiyanarayanan.611702, k.s.kim}@swansea.ac.uk

Abstract—In this paper we propose and investigate the performance of a multi-channel scheduling algorithm based on the well-known deficit round-robin (DRR), which we call multi-channel DRR (MCDRR). We extend the original DRR to the case of multiple channels with tunable transmitters and fixed receivers to provide efficient fair queueing in hybrid time division multiplexing (TDM)/wavelength division multiplexing (WDM) optical networks. We take into account the availability of channels and tunable transmitters in extending the DRR and allow the overlap of 'rounds' in scheduling to efficiently utilize channels and tunable transmitters. Simulation results show that the proposed MCDRR can provide nearly perfect fairness with ill-behaved flows for different sets of conditions for interframe times and frame sizes in hybrid TDM/WDM optical networks with tunable transmitters and fixed receivers.

Index Terms—Multi-channel scheduling, fair queueing, tunable transmitters, hybrid TDM/WDM, quality of service (QoS).

I. INTRODUCTION

Scheduling is a method of harmonizing the access to system resources among competing data flows. It is achieved by specifying the order and the allotted time period for packets from each flow. Scheduling is an important part of networking systems because it not only enables the sharing of the bandwidth but also guarantees the quality of service (QoS). Well-designed scheduling algorithms could provide higher throughput, lower latency, and better fairness with lower complexity in serving the packets. As such, the scheduling plays an important role in achieving high performance of the networking systems.

Due to its importance in networking and communication, the scheduling has been extensively studied but mainly in the context of single-channel communication. The advent of wavelength division multiplexing (WDM) technology, however, demands the extension of this packet scheduling problem to the case of multi-channel communication, especially with tunable transmitters for hybrid time division multiplexing (TDM)/ wavelength division multiplexing (WDM) systems. The main objective of the multi-channel scheduling is to schedule the transmissions of the data over multiple channels to the users. The important measures in choosing a scheduling algorithm are throughput, latency, fairness, and complexity. The major focus of existing work is mostly on the throughput and delay performance of the scheduling algorithm like in SUCCESS-HPON [1], but there is hardly any support for fairness guarantee. The main objective of this paper is to

study the multi-channel scheduling in hybrid TDM/WDM optical networks with tunable transmitters and fixed receivers providing fairness in throughput. In this paper we propose and investigate the performance of a multi-channel deficit round-robin (MCDRR) scheduling algorithm which can provide throughput fairness among flows with different size packets with $O(1)$ processing per packet.

As for multi-channel scheduling with tunable transceivers in hybrid TDM/WDM optical networks, the work for the SUCCESS-HPON architecture in [1] provides a detailed investigation of several multi-channel scheduling algorithms like batching earliest departure first (BEDF) and sequential scheduling with schedule time framing (S^3F) under realistic environments, which is one of the basis for the work in this paper. Through extensive simulations using tunable transmitters and receivers, it has been demonstrated that both the BEDF and S^3F improves the throughput and delay performances. Note that we consider the case of tunable transmitters and fixed receivers in this paper, while the SUCCESS-HPON architecture is based on both tunable transmitters and tunable receivers.

The proposed MCDRR is based on the deficit round-robin (DRR) scheduling algorithm which extends the simple round-robin with deficit counters [2]. The DRR provides good fairness, lower complexity, and lower implementation cost, which makes it an ideal candidate for high-speed gateways or routers.

In the basic DRR scheme, stochastic fair queuing (SFQ) [3] is used to assign flows to queues. For serving the queues, round-robin scheduling is used, with a quantum of service assigned to each flow. The DRR scheduler in rotation selects packets to send out from all flows that have queued. The DRR maintains a service list to keep the flow sequence being served in a round and to avoid examining empty queues. It differs from the traditional round-robin in that if a queue is unable to send a packet in the previous round because a packet was too large, the remainder from the previous quantum is added to the quantum for the next round. Queues that are not completely serviced in a round are compensated in the next round.

During each round, a flow can transmit at once as many packets as possible if there is enough quantum for them. For each flow, two variables — i.e., quantum and deficit counter — are maintained. Quantum is the amount of credits in bytes allocated to a flow within the period of one round. Deciding

the quantum size is an important issue. If we expect that the work for DRR is $O(1)$ per packet, then the quantum for a flow should be larger than the maximum packet size from the flow so that at least one packet per backlogged flow can be served in a round [2].

Note that the multi-channel scheduling have been studied by others in slightly different contexts that ours: For instance, optimal wavelength scheduling for Hybrid WDM/TDM Passive Optical Networks (PONs) [4] inspects the upstream wavelength scheduling in hybrid wavelength division multiplexing and time division multiplexing passive optical networks (WDM/TDM PONs), where the minimum resource allocation unit is a time slot on a wavelength. They use three optimal wavelength scheduling algorithms for the three kinds of hybrid WDM/TDM PONs.

- Type-I WDM/TDM PONs: Each optical network unit (ONU) has a single optical transmitter with a tunable wavelength.
- Type-II WDM/TDM PONs: Each ONU still has a single transmitter, but some are fixed to transmit at a certain wavelength.
- Type-III WDM/TDM PONs: Each ONU has one or more transmitters and all transmitters can tune their wavelengths.

They proposed algorithms based on the round-robin scheduling (RRS) and shortest channel first (SCF) concept to calculate the optimal schedule length and achieve the best wavelength scheduling with the shortest schedule length and the maximum channel utilization. Also, to provide fairness guarantee with multiple channels, the extension of fair queueing (FQ) has been studied in [5], [6], but they are for fixed transceivers as in static WDM systems. The closest to our work in this paper is the study of multi-server round-robin scheduling in [7]. Unlike the hybrid TDM/WDM optical network where a specific wavelength is dedicated to a specific destination, however, they assume that flows can use any of multiple channels.

Our paper is mainly based on the tunable transmitters and fixed receivers in the multi-channel system which requires investigation in the performance of a multi-channel deficit round-robin (MCDRR) scheduling algorithm, which can provide fairness (in terms of throughput) for flows with different size packets with O(1) processing per packet.

The rest of the paper is organized as follows. In Section II we explain the concept of packet service in rounds in the multi-channel case and explain the enqueueing and dequeuing processes in detail in the MCDRR algorithm. We also illustrate the MCDRR algorithm with examples. In Section III we present simulation results for the MCDRR algorithm. Section IV concludes our discussions in this paper.

II. MULTI-CHANNEL DEFICIT ROUND-ROBIN (MCDRR)

The scheduling of packets in switches and routers has been studied mainly in the context of single-channel communication with fixed transceivers. We extend the packet scheduling problem to the case of multi-channel communication with tunable

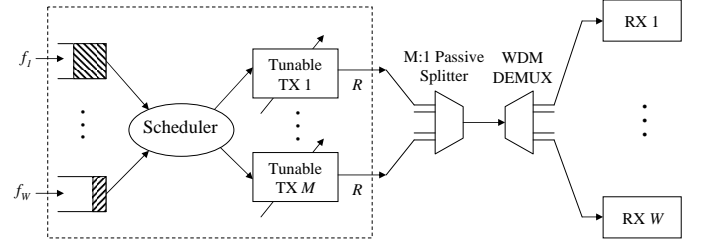transmitters and fixed receivers as shown in Fig. 1.[1] The

Fig. 1. Block diagram of a hybrid TDM/WDM link based on tunable transmitters and fixed receivers.

proposed MCDRR, the multi-channel extension of the DRR, takes into account the availability of channels and tunable transmitters and overlaps 'rounds' in scheduling to efficiently utilize channels and tunable transmitters. To service the queues (i.e., virtual output queues (VOQs)), we use the simple round-robin algorithm with a quantum of service assigned to each queue as in the case of DRR.

Because the MCDRR allows multiple rounds to overlap and run in parallel, the scheduling and the transmission of packets are not necessarily sequential unlike the DRR. To take into account these parallel operations and their timing relations, therefore, we need a precise description of the MCDRR scheduling algorithm and provide a detailed pseudocode in Fig. 2.

$Enqueue(i, p)$ is a standard queue operation to put a packet p into a VOQ for channel i. $Dequeue()$ is a key operation of the MCDRR scheduling and returns a pointer to the head-of-line (HOL) packet in the selected VOQ or $NULL$ when the scheduler cannot find a proper packet to transmit. $packet(queue, pos)$ returns a pointer to the packet at the position of pos in the $queue$ or $NULL$ when there is no such packet.

For each $VOQ[i]$, we maintain the following counters:

- $DC[i]$: It contains the byte that $VOQ[i]$ did not use in the previous round.
- $numPktsScheduled[i]$: It counts the number of packets scheduled for transmission during the service of $VOQ[i]$. Unlike the original DRR, we need this counter to keep track of those packets scheduled for transmission due to multiple rounds overlapped and running in parallel.

A. MCDRR Example

The arrows in the above diagrams shows the triggering of scheduling process after the transmission of packet from each flow. After serving packets from each flow, the tunable transmitter triggers the scheduling process. The Fig. 12 shows the overlapping of rounds. The MCDRR is carried out in such way, where the next round starts as the previous round still in progress. It means that the delay is avoided and the channel does not remain idle when packets satisfy all the criteria.

[1]That is also a model for the downstream links of future hybrid TDM/WDM PON with tunable transmitters at the optical line terminal (OLT) and fixed receivers at the optical network units (ONUs).

Initialization;
for $i \leftarrow 0$ **to** $W - 1$ **do**
    $DC[i] = 0$;
end

Arrival *on the arrival of a packet p from channel i*;
if *Enqueue(i, p) is successful* **then**
    if *a transmitter is available* **then**
        $(ptr, \ ch) \leftarrow Dequeue()$;
        if $pkt \neq NULL$ **then**
            $Send(*ptr, \ ch)$;
            if *VOQ[i] is empty* **then** $DQ[i] \leftarrow 0$;
        end
    end
end

Dequeue;
$startQueueIndex \leftarrow (currentQueueIndex + 1)\%W$;
for $i \leftarrow 0$ **to** $W - 1$ **do**
    $idx \leftarrow i + startQueueIndex\%W$;
    if $VOQ[idx]$ *is not empty* **then**
        $DC[idx] \leftarrow DC[idx] + Q[idx]$;
        if $numPktsScheduled[idx] == 0$ **then**
            $currentQueueIndex \leftarrow idx$;
            $pos \leftarrow 0$;
            $ptr \leftarrow \&packet(VOQ[idx], \ pos)$;
            repeat
                $DC[idx] \leftarrow DC[idx] - length(*ptr)$;
                $numPktsScheduled[idx] + +$;
                $pos + +$;
                $ptr \leftarrow \&packet(VOQ[idx], \ pos)$;
                if *ptr is NULL* **then** Exit the loop;
            until $DQ[idx] \geq length(*ptr)$;
            $Return \ (\&packet(VOQ[idx], \ 0),$
                $currentQueueIndex)$;
        end
    end
end
$Return \ NULL$;

Departure *at the end of transmission on channel i*;
$numPktsScheduled[i] - -$;
if $numPktsScheduled[i] > 0$ **then**
    $ptr \leftarrow \&packet(VOQ[i], \ 0)$;
    $Send(*ptr, \ i)$;
    if *VOQ[i] is empty* **then** $DC[i] \leftarrow 0$;
end
else
    $(ptr, \ ch) \leftarrow Dequeue()$;
    if $ptr \neq NULL$ **then**
        $Send(*ptr, \ ch)$;
        if *VOQ[ch] is empty* **then** $DC[ch] \leftarrow 0$;
    end
end

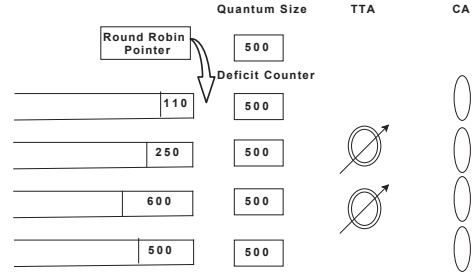Fig. 2.   Pseudocode for the MCDRR algorithm.
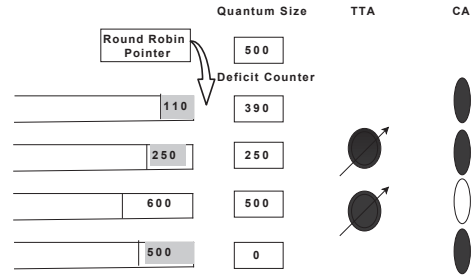

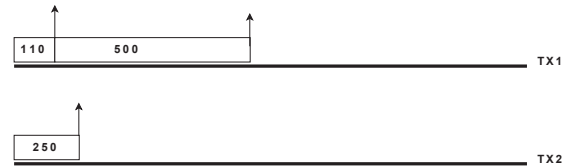Fig. 3.   Start of Round 1

Fig. 4.   End of Round 1

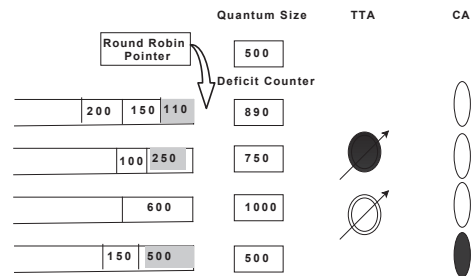Fig. 5.   End of Round1 showing in Tunable Transmitter

Fig. 6.   Start of Round 2
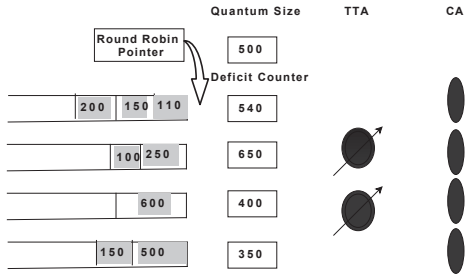
Fig. 7. End of Round 2

Fig. 8. End of Round2 showing in Tunable Transmitter

Fig. 9. Start of Round 3

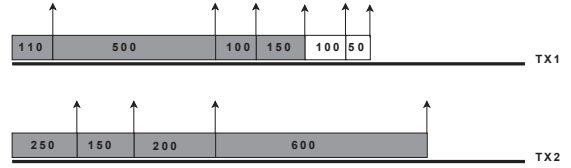Fig. 10. End of Round 3
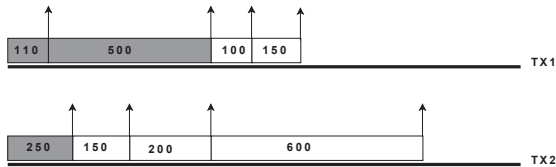

Fig. 11. End of Round3 showing in Tunable Transmitter

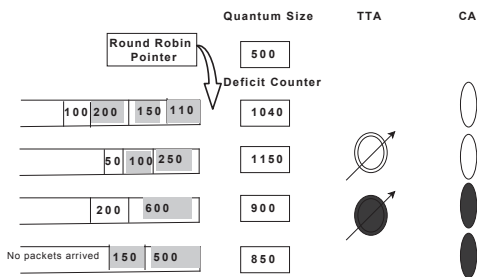Fig. 12. Overlapping of all the three rounds in MCDRR Scheduling
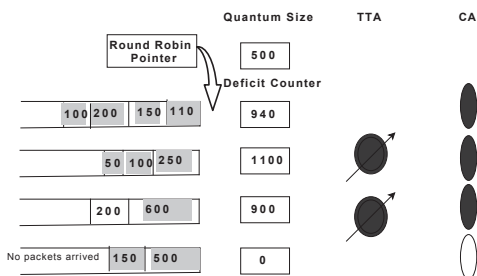
B. MCDRR Example Explanation

At the start of the First Round, the tunable transmitter available triggers the scheduling process. The round robin pointer starts from the first flow initialized. The deficit counter becomes equal to the quantum size. If the packet size is lesser than the deficit counter and channel is available at that instant of time, the packet is served. If the channel is not available, the pointer is moved to the next flow. When the channel becomes available then the packet will be transmitted in the next round.

In the example quantum size is considered to be 500 credits, now both the tunable transmitters are available, the pointer starts from the Flow 1, the packet of size 110 bytes will be served since it is less than the deficit counter 500 credits and the channel is available at that instant of time. By default they choose tunable transmitter 1. After serving, the deficit counter is updated, that is DC becomes 390 credits.

Since the tunable transmitter 2 is also available, the pointer moves to the Flow 2, the packet of size 250 bytes will be served since they are less than the deficit counter 500 credits and the channel is available at that instant of time. DC is updated. Now TX1 becomes available and triggers the scheduling process, the pointer moves to Flow 3, the packet size is greater than the deficit counter and the flow is skipped. DC remains the same. Still the TX1 is available, so the packet in Flow 4 of size 500 bytes is served successfully. Since the pointer has moved through all the given flows, we say it as

"Completion of one Round".

Now TX2 becomes available and triggers the scheduling process, which is the start of next Round, that is Second Round. The deficit counter is updated with the quantum size again i.e quantum is added to all the deficit counters of the respective flows. DC= DC(prev) + Quantum Size. The pointer starts from the Flow 1 again. The DC becomes 390 credits + 500 credits. In this second round, two packets in Flow 1 had arrived. According to our description, only one packet can be served from each flow irrespective of packet size as far as they satisfy the dequeuing criteria. So the packet size of 150 bytes can be served successfully with channel available at that instant but not the packet of 200 bytes (because only one packet can be served per flow as per our description). After the service, the DC is updated again.

At some instant, both the tunable transmitters TX1 and TX2 can be available. In that case by default TX1 will be chosen. In this case TX2 triggers again, the packet of 100 bytes in Flow 2 is served successfully. The packet in Flow 3 which was not served in previous round is been served in this round using TX2 because the DC is 1000 credits now. Now TX1 becomes available and the packet of 150 bytes in Flow 4 is served, that is the End of Round.

The TX1 becomes available and triggers the scheduling process, which is the start of the next round, that is the Third Round. The DC is updated with the quantum size again. The pointer starts from the Flow 1. The packet of 200 bytes in the Flow 1 is served and after some instant again TX1 becomes available and the packet in Flow 2 is served. Since the packet size is small, TX1 becomes available at the earliest compared to the TX2. The packet (200 bytes) in Flow 3 cannot be served at this instant though the tunable transmitter is available because channel is not available, it means that the packet is still being served from the previous round. So the pointer moves to the next flow and packets arrived in this round means no packets to be served in this Third round i.e Flow is empty, in that case DC is reset to zero for fairness issues. So that is the end of this round. Since TX1 is still available, the next round starts from the Flow 1 and the process continues till all the flows completely become empty which is sequential.

This example covers all the details such as packet size lesser than the deficit counter with channel available and channel not available at some instant of time, then packet size greater than the deficit counter with channel available and not available, flow being empty in one particular round.

III. Simulation Results

Fig. 13 (a) and (b) show the throughput for 16 flows for two different sets of conditions for inter-fame times and frame sizes.

To demonstrate the performance of the proposed MCDRR scheduling algorithm, we carried out simulation experiments with a model for a hybrid TDM/WDM link with tunable transmitters and fixed receivers shown in Fig. 1.

We set the number of wavelengths/channels (W), the line rate of each channel, and the number of tunable transmitters

(M) to 16, 1 Gb/s, and 2, respectively. We assume that the scheduling is done at the data link layer with Ethernet frames and ignore the tuning time of tunable transmitters in simulation. Each VOQ can hold up to 1000 frames. We measure the throughput of each flow at a receiver for 10 mins of simulation time.

Fig. 13 (a) and (b) show the throughput for 16 flows for two different sets of conditions for inter-fame times and frame sizes.

In Fig. 13 (a), the interframe times are exponentially distributed with the averages of 16 μs and 48 μs for the first flow and the rest of the flows respectively, while the frame sizes are uniformly distributed between 64 and 1518 bytes for all the flows. In Fig. 13 (b), the interframe times are exponentially distributed with the averages of 16 μs and 32 μs for the first flow and the rest of the flows, while the frame sizes are fixed to 1000 bytes for the first flow and 500 bytes for the rest of the flows. For both the cases, the first flow sends frames at four times the rate of other flows. The combined traffic rates[2] are 2.409 Gb/s for Fig. 13 (a) and 2.375 Gb/s for Fig. 13 (b), which slightly overload the link. Raj Jain's fairness index [8] for the results of Fig. 13 (a) and (b) are 0.9999756 and 0.9999998, respectively.

From the simulation results, we found that the proposed MCDRR scheduling algorithm provides nearly perfect fairness even with ill-behaved flows for different sets of conditions for interframe times and frame sizes.

IV. Conclusion

In this paper we have proposed and investigated the performance of the MCDRR scheduling algorithm for a multi-channel link with tunable transmitters and fixed receivers, which is based on the DRR, the well-known single-channel scheduling algorithm. In extending the DRR to the case of multi-channel scheduling, we try to efficiently utilize the network resources (i.e., channels and tunable transmitters) by overlapping rounds, while maintaining its low complexity (i.e., $O(1)$). The nearly perfect fairness provided by the MCDRR has been demonstrated through simulation experiments. Establishing mathematical bounds for the fairness and latency of the MCDRR and comparison with other multi-channel scheduling algorithms are now under study.

V. Acknowledgment

References

[1] K. S. Kim, D. Gutierrez, F.-T. An, and L. G. Kazovsky, "Design and performance analysis of scheduling algorithms for WDM-PON under SUCCESS-HPON architecture," *J. Lightw. Technol.*, vol. 23, no. 11, pp. 3716–3731, Nov. 2005.
[2] M. Shreedhar and G. Varghese, "Efficient fair queueing using deficit round robin," *SIGCOMM Comput. Commun. Rev.*, vol. 25, no. 4, pp. 231–242, 1995.
[3] P. E. McKenney, "Stochastic fairness queueing," *Internetworking: Research and Experience*, vol. 2, pp. 113–131, Jan. 1991.

[2]The inter-fame gap (IFG) of 12 bytes is taken into account.
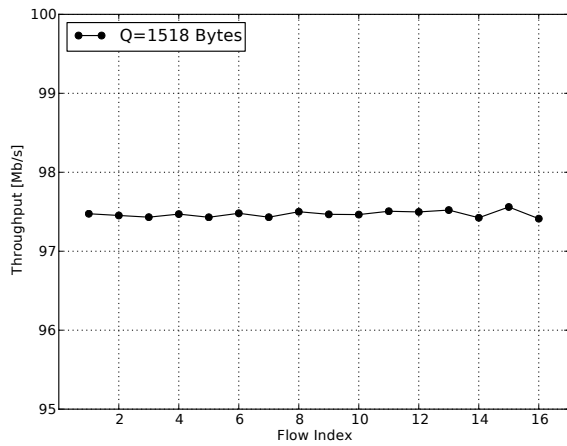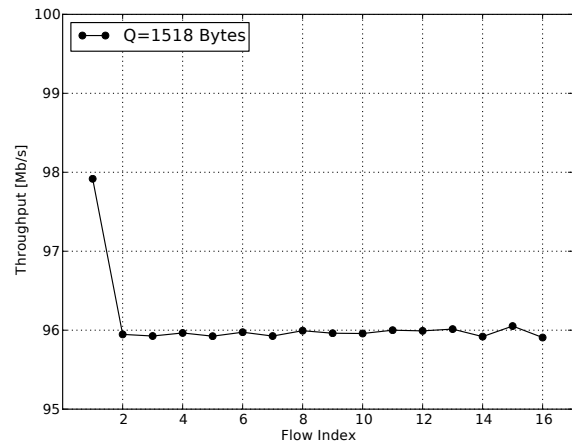
Fig. 13. Throughput for 16 flows with (a) exponential interframe times and random frame sizes and (b) exponential interframe times and fixed frame sizes.

[4] C. Wang, W. Wei, W. Zhang, H. Jiang, C. Qiao, and T. Wang, "Optimal wavelength scheduling for hybrid WDM/TDM passive optical networks," *J. Opt. Commun. Netw.*, vol. 3, no. 6, pp. 522–532, Jun. 2011.

[5] J. M. Blanquer and B. Özden, "Fair queuing for aggregated multiple links," in *Proc. of SIGCOMM'01*, San Diego, CA, USA, Aug. 2001, pp. 189–197.

[6] J. A. Cobb and M. Lin, "A theory of multi-channel schedulers for quality of service," *Journal of High Speed Networks*, vol. 12, no. 1-2, pp. 61–86, 2002.

[7] H. Xiao and Y. Jiang, "Analysis of multi-server round robin scheduling disciplines," *IEICE Trans. on Commun.*, vol. E87-B, no. 12, pp. 3593–3602, Dec. 2004.

[8] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," Digital Equipment Corporation, Tech. Rep. DEC-TR-301, Sep. 1984.