Submitted to International Conference on Telecommunications - ICT 2002

Avoiding Head-of-Line Blocking using an Enhanced Fairness Algorithm in a Resilient Packet Ring

Stein Gjessing and Fredrik Davik

Simula Research Laboratory / University of Oslo

P.O. Box. 134 Lysaker, 1325 Lysaker, NORWAY

Email: {steing, bjornfd}@simula.no

Abstract. **IEEE is currently conducting an effort to standardize a full duplex, spatial reuse, ring network architecture, called the Resilient Packet Ring (IEEE P802.17). We discuss the relationship between spatial reuse and head of line blocking in such a ring. An algorithm that schedules packets individually based on destination address is outlined. We have written a model of the RPR architecture in the programming language Java, and simulated several scenarios in order to evaluate and compare our algorithm with a conventional ring fairness algorithm. Our performance evaluations show that it is possible to achieve significant higher throughput for traffic scenarios where head of line blocking traditionally has degraded performance. Where head of line blocking is not an issue, our algorithm behaves like the traditional one.**

I. Introduction

The Resilient Packet Ring (RPR) is a scalable high performance network architecture that connects nodes or stations into a point-to-point, full duplex, ring topology. RPR is currently under standardization by the IEEE working group P802.17. The goal of the working group is to define a standard that can be used for different sized rings and high data rates. The original goal of the IEEE 802 committee is to standardize LANs and MANs, but for 802.17 also Wide Area Network usage is considered. RPR should be able to utilize underlying technologies of various types, e.g. WDM, SONET and high-speed point-to-point Ethernet.

The full duplex (one or more ringlets in each direction) ring topology has several nice properties. A station can chose onto which of the rings it will place a packet, and hence minimize the travel distance (and possibly also the congestion) from source to destination.

A dual ring is fault tolerant [1,21]. If a segment of the ring breaks, the immediate nodes on each side can wrap the traffic around and send the packets on the longer, but available, path around the ring. When a station learns about a broken segment, it can change the direction of packet insertion. However, notice the reduced aggregate bandwidth of a broken ring.

Ring topologies have been popular for LANs and MANs for a long time. The Cambridge Ring was designed as early as in the mid 1970's [18], while the IEEE Token Ring standard [10] and FDDI [21] were developed later. In the early systems, access to the ring was regulated by a token, but later spatial reuse was exploited, e.g., in systems like MetaRing [6], ATMR [13], CRMA-II [15], DQDB [11] and SCI [12]. The access methods for these rings have been extensively studied, compared and refined [2,5,9,16,17,20].

Unlike a token ring, where only one packet uses the ring at a time, a destination removal ring like RPR can have several packets on their way at the same time, provided the packets use different segments of the ring. This is called spatial reuse and is illustrated on the inner ring on figure 1. Packets that are traveling on the ring and passing by a station will have to wait if a locally sourced packet is being transmitted. While waiting, the packet is placed in what we call the ***transit*** buffer. Each station also contains a set of ***ingress*** buffers where packets wait to be transmitted onto the ring, and a set of ***egress*** buffers where packets are placed when they are removed from the ring.

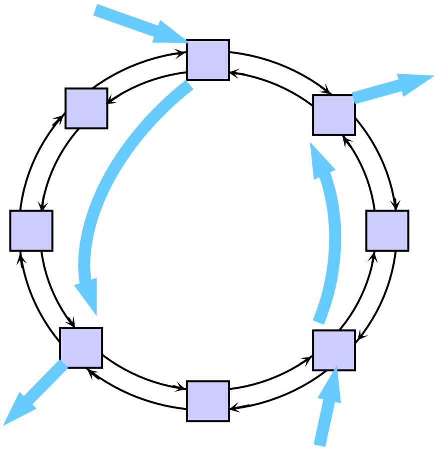

Figure 1. RPR-ring - Spatial reuse shown on the inner ring

A very naïve algorithm for ring access would be for each station to start transmitting a packet as soon as the transit buffer is empty. This would, however, make it easy for one station to starve downstream neighbors by filling the ring completely. In order to avoid such starvation and give each station fair access to the ring, a fairness algorithm must be deployed [4,5,12,23].

Most of the fairness algorithms rely on telling the upstream node to send idle symbols or empty packets. In particular, [4] discuss an advanced method for tracking packets and achieving good spatial reuse in this way. The fairness algorithm used by Cisco, called Spatial Reuse Protocol (SRP) [23], is based on a regulation scheme where each congested station sends information about its recent transfer rate upstream. In a stable situation, all upstream stations will then transmit onto the ring the same amount as the downstream stations.

It is, however, not always optimal for all stations to send an equal amount of data. This has been discussed for SCI [20], and MetaRing [4]. The latter paper discusses some of the problems presented in this paper, but solves the problem using a different approach.

In this paper we present and discuss a fairness algorithm that utilizes ring bandwidth significantly better by HOL blocking avoidance. The possibility for better spatial reuse and our enhanced fairness algorithm is the topic of the next section. Then we present the ring model used in our evaluation. Three traffic scenarios are discussed, and we show how a conventional fairness algorithm and our new fairness algorithm behave in these cases. Finally, we summarize and conclude our findings.

II. AN ENHANCED FAIRNESS ALGORITHM

One of the main advantages of destination removal is the possibility to send packets concurrently on different ring segments. However, when several stations are waiting to use the link, spatial reuse may be reduced. In particular, this will be the case when the ingress queues are strictly FIFO. Figure 2 is used to illustrate this. Stations 0 through 6 send data to station 7. Hence, they all require link bandwidth along the path from source to destination. In particular, they all need to send data on the link between stations 6 and 7. As an example, assume that the first packet in the ingress buffer of station 0 is destined for station 7 and the second packet is destined for station 1. While station 0 is waiting to send to station 7, the link from 0 to 1 is unused. The second packet is however blocked by the packet waiting at the head of the line, thus the opportunity to utilize the unused bandwidth from station 0 to station 1 is lost (no spatial reuse). This is called head of line (HOL) blocking. The same HOL blocking situation will also happen, but to a lesser extent (lesser unused bandwidth), at stations 1 to 5.

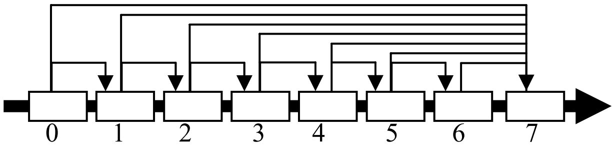

Figure 2. Hot receiver plus local traffic

There are at least two factors that must be fulfilled to optimize utilization of unused link bandwidth and avoid HOL blocking:

1. In each station there must be a separate ingress buffer per destination station (or another mechanism to the same effect)
2. The fairness algorithm must know the status of each downstream link and/or transit buffer.

A cost and complexity analysis of buffer technologies required to avoid HOL blocking is outside the scope of this paper. This paper discusses enhanced algorithms and evaluates possible performance improvements in rings with non-HOL blocking buffers.

As noted by 2), a fairness algorithm which avoids HOL blocking, must know the situation in the downstream parts of the ring. The algorithm must be able to decide when it is advisable and fair to send a packet to a station far away on the ring, and when it may transmit a packet that has a more immediate downstream station as its destination only.

All links connect two stations. The upstream station is called the ***owner*** of the link. We suggest an algorithm where all owners collect status information about their links. (We focus on one of the rings transporting data unidirectionally; the algorithm for the ring sending data in the other direction will be congruent). Each station sends its status data in a control packet upstream (on the link going in the other direction). All upstream stations copy the content of this packet. It is a broadcast packet that circulates the ring once. When the packet returns to the sender, it is stripped from the ring. Alternatively, one could collect all status data from all stations in one (or a few) packet(s) that circulate on the ring. The bandwidth used by such control packets is low. Assuming 32 byte packets are used and each station sends out its own status packet every 50 000 byte counts, this result in a 1% bandwidth usage for control packages in a 16-node network.

When a station has received control packets from all downstream nodes, it has global knowledge of the status of the ring. This information is used by the station to inhibit sending of data exceeding the nodes fair share of the ring bandwidth from its ingress buffer. The fairness algorithm basically has two possibilities:

I) The station inhibits sending of new packets onto the ring for the time it takes the downstream transit buffer(s) to empty (or its content size is reduced below a certain threshold). This is the approach taken by MetaRing [6] and SCI [12]

II) The station inhibits sending to adjust to the sending rate of the downstream neighbor(s). This is the approach taken by SRP [23].

In the implementation of the enhanced fairness algorithm reported in this paper, we use a variant of method II.

Each station keeps track of its send rate, i.e. the number of packets (bytes/sec) it has transmitted (from its ingress buffers) onto the ring lately. In order to avoid high frequency oscillations, this information is run through a low pass filter. When a station is congested (it is not able to send as much data onto the ring as it wants to), it advertises its send rate to all upstream neighbors in a control packet. All upstream neighbours store this value from station i in a counter o_i. In addition, all stations have a set of counters, s_i , which tracks their bandwidth usage on all downstream links i. The s_i counters are updated for all links between sender and destination each time a packet is transmitted.

Hence, all stations keep two counters for each downstream link i: The o_i counters are the owners' send rate on link i. The values of these counters are copied from the broadcasted control packets. The s_i counters tracks this stations bandwidth usage on the downstream link i.

Based on these counters, our enhanced algorithm decides whether the station may send another packet, and in particular how far downstream a packet may be sent at this time. For link i, this means that as long as the stations send rate over a link i is less than that of the owners send rate over this link (i.e. as long as $s_i < o_i$), the station may send another packet over this link.

When a station i is not congested, is does not advertise its send rate, i.e. no control packets are sent from this station. The fairness algorithm allows the upstream stations to gradually increase their o_i values in an attempt to send more. They will be able to do so until station i gets congested, at which point station i again advertises its send rate, resulting in a decrease in the o_i values of the upstream stations, etc.

In a non-HOL blocking ingress buffer, several fairness principles can be used in order for the algorithm to choose among the packets it is allowed to send. In the implementation evaluated in the sequel, we have adopted a round robin strategy among all destination addresses.

III. THE EVALUATION METHOD

We use performance evaluation by simulation to support our comparisons and discussions. We have designed a model of the RPR-ring and implemented it in the programming language Java. Using our ring model we can simulate different technology solutions, different link speeds and ring sizes and different traffic scenarios. The tractable model size (a few thousand lines of Java code) makes it easy to modify and vary any parameter and all aspects of the ring. The ability to write results to file from anywhere in the program, provides the opportunity to extract the information needed from our executable model.

In order to compare the capabilities of our enhanced fairness algorithm with those of a conventional one, we chose to implement SRP [23] in addition to our own enhanced fairness algorithm. One of the important reasons for implementing SRP, is that it is well defined. SRP does not solve the HOL blocking problem, i.e. the destination address of the packet is not a parameter when a station has to decide whether it can send a packet.

Our packets have two levels of priority. Control packets are sent with high priority, while data traffic has low priority. When a station is choosing a new packet to send out on its link, it first looks for a high priority packet in the transit buffer (there are separate transit buffers for high and low priority packets). If that buffer is empty, it looks for high priority packets in the ingress buffer. If there are no high priority packets to send, data from the low priority transit and ingress buffers have equal priority (it keeps an equal byte count). This equal priority is in effect only as long as the size of the transit buffer is below a threshold. When the size is above this threshold, the transit buffer has priority. The fairness algorithms do not control high priority traffic. Hence, a large amount of high priority traffic could make the low priority transit buffers overrun. A discussion of this is outside the scope of this paper.

The results shown in the sequel are for rings of size 16 stations, but we have run the experiments with 32, 64, and 128 stations as well, and the results are the same. Packets are sent on the shortest path to the destination. When a packet is destined to the station directly across from the sender, the outer ring is chosen.

It takes one clock tick to send a symbol (a byte) out on a link. The propagation delay of each link is 2500 clock ticks. This latency includes the time it takes to pass through an empty station, i.e., a station where the transit buffer is empty. Waiting time in none-empty transit buffers is included in the simulation results. Assuming a clock tick of three nanoseconds, the simulation results are valid for a system with a link speed of just above 2.5Gbit/sec (OC-48) and almost 1.5 km links. We have conducted some of the experiments below with longer links as well (up to 50000 clock ticks), and the results we saw then are the same as we see here for shorter links.

For simplicity, we have chosen to use one packet size. A size of 500 bytes has been shown to be a good choice for a “typical” data packet [7]. The control packet size is 32 bytes.

All experiments have been conducted with long simulation times and different seeds, and the results shown are averages for steady state. For experiments A and B below the results shown are values for 10^8 ticks (OC-48: 300 ms). The final standard deviation is small. The 99 % confidence interval for the average would hardly be visible in our plots.

IV. THE EXPERIMENTS

Three main experiments were set up to demonstrate the effect of our enhanced fairness algorithm. The first one is a hot receiver scenario, the second a hot sender and the last one is a random traffic scenario. As performance is mainly an issue when the system is heavily loaded, all senders in all scenarios are pushing as much data as possible onto the ring.

A. Hot receiver scenario

The advantages of our new algorithm is best seen when HOL blocking effects are most serious. We have implemented the example described by figure 2: Assume station 7 is the rings connection to the outside, and that stations 0 through 6 are streaming as much data as possible out of the ring, i.e. to station 7. In addition, stations 0 through 6 also stream as many packets as possible to their immediate downstream neighbor. In figure 3 (Hot receiver – Traditional fairness) we see the effect of a traditional fairness algorithm that does not consider the destination address of the packages. The farthest downstream sender station (station 6), sends a number of packets to the hot receiver (station 7). This number gets propagated upstream to station 5, allowing it to send out approximately the same number of packets. Because of HOL blocking, station 5 transmits on average 50% to station 7 and 50% to station 6. Going further upstream, stations 4, 3, 2 and 1 sends in the same way as station 5. The only station that gets to send more is station 0.

We have not fully understood the reason for this, but it is probably because of the spatial reuse caused by the packets traveling only one hop. This causes the transit buffers in nodes 1 through 5 to only carry traffic to the hot receiver, not any local traffic. This “fools” station 0 to believe it can get more bandwidth than stations 1 to 5. We have repeated the experiment with more stations on the ring, and the same pattern is observed.

Figure 4 shows the same scenario with our new algorithm installed. We see that all stations get the same fair amount of bandwidth to the hot receiver. This is possible because station 6 sends control packets upstream to all stations telling them how much station 6 itself is using of the bandwidth between stations 6 and 7. The other stations then adjust to this value when they send packets to station 7. When they send to the other stations they ignore this value. Hence, they send as much as they can to their immediate downstream neighbor. We also see that all links between stations 0 to 7 are fully utilized. E.g. for station number 3, the “Sent to immediate downstream neighbour” value is 83692. In addition, on the link from station 3 to station 4, also the “Sent to hot receiver” values from stations 0, 1, 2, and 3 are transmitted. These values are respectively 28341, 31284, 28342 and 28340. The sum of these values is 199999 which equals the full link bandwidth:

$$\frac{\text{length of observation period[ticks]}}{\text{packet transmit time[ticks/packet]}} = \frac{10^8}{500}\text{[packets]} = 200.000\text{[packets]}.$$

For the traditional fairness algorithm (figure 3), the link utilization is limited between stations 0 – 6. The link is fully utilized only between stations 6 and 7. E.g. when considering the link betweens stations 3 and 4, the aggregate throughput is reduced to 132.334 [packets] (calculated as above).

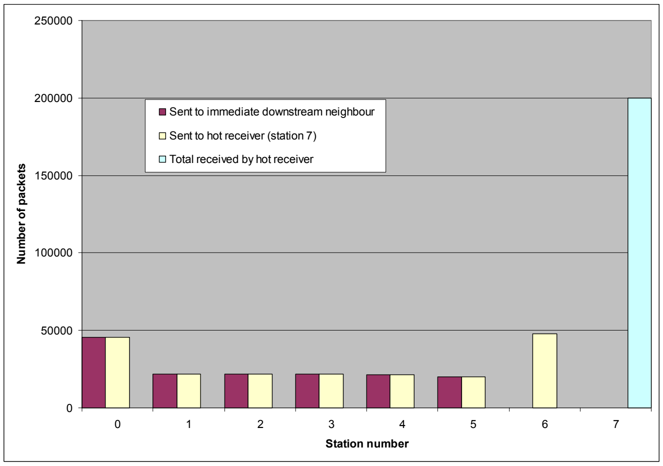

Figure 3. Hot receiver – Traditional fairness

B. Hot sender scenario

In this scenario we assume that station 0 is the connection to the outside, and that it receives seven streams of data that it passes on to stations 1 through 7. At the same time stations 1 through 6 sends as much data locally as possible to their immediate downstream neighbor. Since these stations (1 through 6) each send one stream only, they experience no HOL blocking.

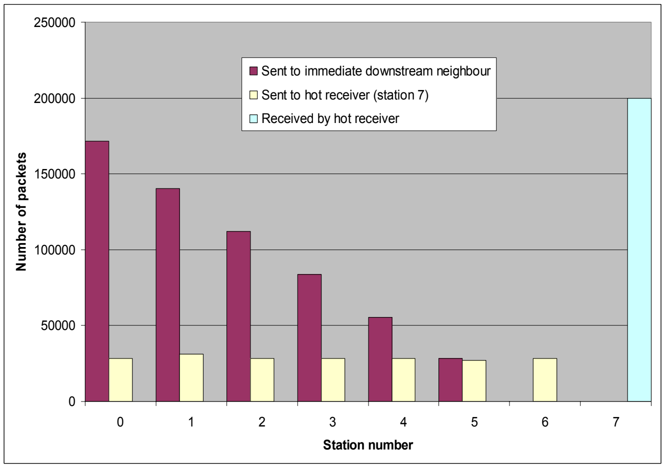

Figure 4. Hot receiver – Enhanced fairness.

In figure 5 (Hot sender – Traditional fairness) we see that with a traditional fairness algorithm and a HOL blocking queue in station 0, each of the stations gets equally many packets from station 0. The traffic between the pairs 1 to 2, 2 to 3, etc. is utilizing all available bandwidth. The only under-utilized link is the one between stations 0 and 1. Hence, we get relatively good spatial reuse even with the traditional algorithm.

Looking at figure 6 (Hot sender – Enhanced fairness) we see that with an ingress buffer in station 0 that is not blocking the head of the line, our new algorithm is capable of utilizing the free bandwidth from station 0 to station 1. It can do so because it knows about the free bandwidth on this link, and can treat the packets destined for station 1 specifically. Now all links are fully utilized.

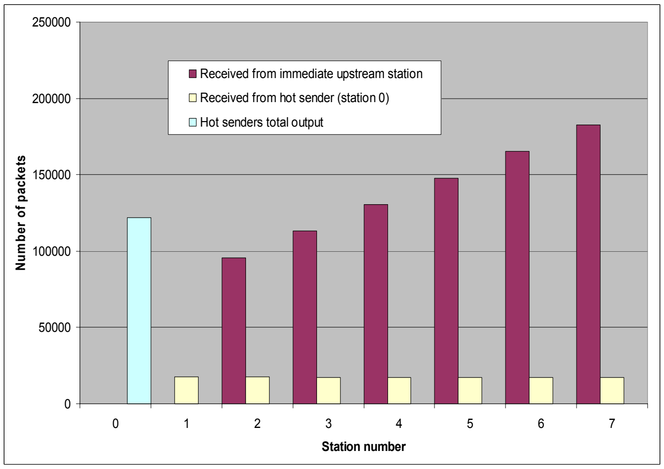

Figure 5. Hot sender –Traditional fairness

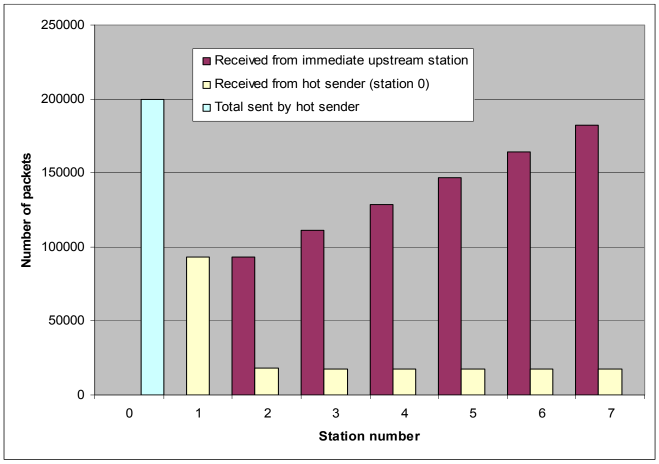

Figure 6. Hot sender – Enhanced fairness

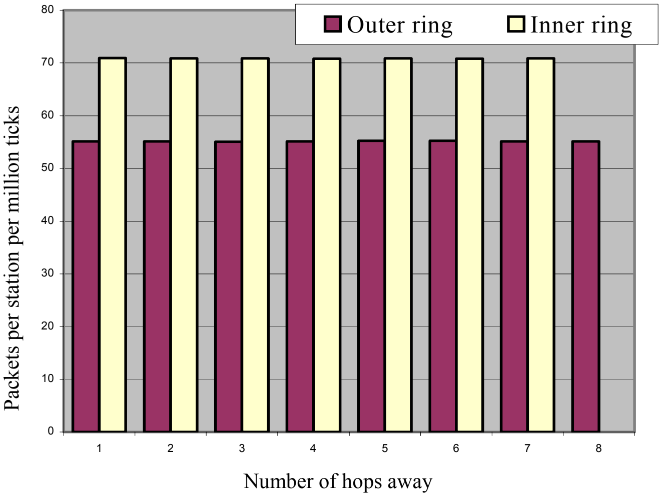

Figure 7. – Random traffic – Traditional fairness

C. *Random traffic scenario.*

In order to compare the two algorithms when there is no special opportunity for extra spatial reuse, we let all stations send packets randomly to all other stations. This is also the only case reported in this paper where the ring is fully loaded in both directions, and where control and data packets compete about access to the links. Each of the 16 stations in the ring sends packets randomly to all of the 15 other stations.

The results are shown in figure 7 and 8. The plots show the average number of packets a station sends to downstream stations 1 through 8 hops away. Since packets to the station directly across from the sender uses the outer ring, the average packet travel distance on this ring is 4.5 hops, while the average packet travel distance on the inner ring is 4 hops. This means that each station on average shares the capacity of its outgoing link on the inner ring with its 3 closest upstream neighbours, leaving the node with ¼th of the link capacity to use for sending packets from its ingress buffer. Thus, during one million ticks, a station can at most send 500 packets (à 500 byte) from its ingress buffer onto the ring. From figure 7 and 8, we find the total number of packets sent on the inner ring is slightly higher than 7x70 = 490. This means that the inner ring is almost fully utilized. The same argument applies to the outer ring.

There is not much difference in the results for the enhanced and the traditional algorithm. The traditional algorithm has a slightly higher overall data throughput. Given that our algorithm uses slightly more control data this is not surprising.

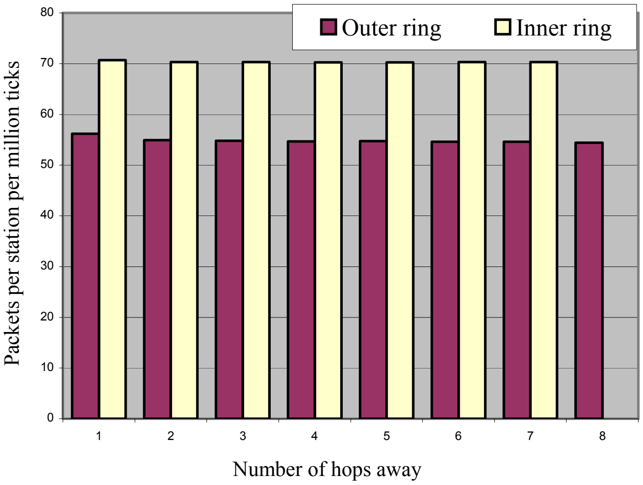

Figure 8. Random traffic – Enhanced fairness

V. CONCLUSION

In an RPR-like ring with HOL blocking ingress buffers and a conventional fairness algorithm, spatial reuse is not fully exploited. We have devised an enhanced fairness algorithm that avoids HOL blocking in a full duplex ring. In order to evaluate our new enhanced algorithm, we have implemented it in a ring model together with a conventional one.

Our experiments have shown that non-HOL blocking ingress buffers combined with our new algorithm, increase spatial reuse significantly for some traffic scenarios. In addition it performs equally well when HOL blocking is not an issue. Our enhanced spatial reuse algorithm is not much more complex than a traditional one, and the overhead of control packets is not significantly more either.

Our approach has an advantage whenever a station has traffic for more than one destination. The algorithm is dynamic, i.e., the transmit pattern need not be known in advance. The real advantage of our approach remains to be seen, and will depend upon real traffic patterns in deployed communication rings.

In follow-up work we would like to evaluate the algorithm under more dynamic traffic patterns. We will also look into the possibility of aggregating more status information in each packet and send them hop by hop (instead of broadcast around the complete ring).

Acknowledgment

The author thanks Olav Lysne for his contribution to the simulator and his valuable comments on this paper.

References

1. ANSI T1.105.01-2000: Synchronous Optical Network (SONET) - Automatic Protection.
2. H.R. van As: Major Performance Characteristics of the DQDB MAC Protocol. Telecommunications Symposium, 1990. ITS'90 Symposium Record, SBT/IEEE 1990
3. S. Breuer, T.Meuser: Enhanced Throughput in Slotted Rings Employing Spatial Slot Reuse. INFOCOM '94. Networking for Global Communications. IEEE. 1994
4. I. Cidon, L. Georgiadis, R. Guerin, Y. Shavitt: Improved fairness algorithms for rings with spatial reuse. IEEE/ACM Transactions on Networking, Vol.5, No. 2, 1997.
5. I. Cidon, Y. Ofek: Distributed Fairness Algorithms for Local Area Networks with Concurrent Transmissions. In: Lecture Notes in Comp. Sci., Vol. 392, Springer, 1988
6. I. Cidon, Y.Ofek: MetaRing - A Full-Duplex Ring with Fairness and Spatial Reuse. IEEE Trans on Communications, Vol. 41, No. 1, January 1993.
7. K.C. Claffy: Internet measurements: State of DeUnion. http://www.caida.org/outreach/presentations/Soa9911
8. M.W. Garrett, S.-Q. Li: A study of slot reuse in dual bus multiple access networks**.** IEEE Journal on Selected Areas in Communications, Vol. 9 Issue 2, Feb. 1991
9. A. Grebe, C. Bach: Performance comparison of ATMR and CRMA-II in Gbit/s-LANs. SUPERCOMM/ICC '94, IEEE Int. Conf. on Serving Humanity Through Communications, 1994
10. IEEE Standard 802.5–1989, IEEE standard for token ring
11. IEEE Standard 802.6–1990, IEEE standard for distributed queue dual bus (DQDB) subnetwork
12. IEEE Standard 1596–1990, IEEE standard for a Scalable Coherent Interface (SCI)
13. ISO/IECJTC1SC6 N7873: Specification of the ATMR Protocol (V. 2.0), January 1993
14. I. Kessler, A. Krishna: On the cost of fairness in ring networks. IEEE/ACM Trans. on Networking, Vol. 1 No. 3, June 1993
15. W.W. Lemppenau, H.R.van As, H.R.Schindler: Prototyping a 2.4 Gbit/s CRMA-II Dual-Ring ATM LAN and MAN**.** Proceedings of the 6th IEEE Workshop on Local and Metropolitan Area Networks, 1993.
16. M.J. Marsan et al.: Slot Reuse in MAC Protocols for MANs. IEEE J. on Selected Areas in Communications. Vol. 11, No. 8, October 1993.
17. H.R. Muller et al: DQMA and CRMA: New Access Schemes for Gbit/s LANs and MANs. INFOCOM '90, Ninth Annual Joint Conference of the IEEE Computer and Communication Societies. IEEE , 1990
18. R.M. Needham, A.J. Herbert: The Cambridge Distributed Computing System. Addison-Wesley, London, 1982.
19. T. Okada, H. Ohnishi, N. Morita: Traffic control in asynchronous transfer mode. IEEE Communications Magazine , Vol. 29 Issue 9, Sept. 1991
20. D. Picker, R.D. Fellman: Enhancing SCI's fairness protocol for increased throughput. IEEE Int. Conf. On Network Protocols. October, 1993.
21. F.E. Ross: Overview of FDDI: The Fiber Distributed Data Interface. IEEE J. on Selected Areas in Communications, Vol. 7, No. 7, September 1989.
22. I. Rubin, H.-T. Wu: Performance Analysis and Design of CQBT Algorithm for a Ring Network with Spatial Reuse. IEEE/ACM Trans on Networking, Vol. 4, No. 4, Aug. 1996.
23. D. Tsiang, G. Suwala: The Cisco SRP MAC Layer Protocol. IETF Networking Group, RFC 2892, Aug. 2000