for the next metro IP networks.

Unlike token ring [4], the data transmission through an RPR node does not depend on whether it holds the token or not and the RPR provides destination release of the data traffic. Therefore, spatial reuse [5]-[8] can be achieved in RPR to better utilize the ring bandwidth. Unfortunately, the ability of the most current RPR algorithms [9 - 11] to achieve their desired objectives (fairness, high utilization and spatial reuse) simultaneously is limited.

A known problem in aforementioned Medium Access Control (MAC) protocols is that a node may starve its downstream nodes. To solve this problem, the RPR standard defines a fairness policy that the upstream nodes must inject traffic at a rate according to the downstream congestion situation. Two well-known modes, namely Aggressive Mode (AM) [6], [10], [11] and Conservative Mode (CM) [9], [11], are used in most RPR algorithms. However, the mechanism for assigning rates at each of the modes is not optimal and may lead to severe oscillations and hence performance degradation. Inheriting the mechanisms of AM and CM, the distributed virtual-time scheduling in ring (DVSR) [7, 8, 12] scheme achieves better performance than AM and CM by better rate allocation assignment.

All these algorithms have one common mechanism that the rate adjustments are controlled by feedback based on reaction to congestion somewhere, so why not make the adjustment as early as possible instead of waiting for the congestion to occur. Obtaining such information earlier will reduce oscillations and achieve better performance [13]. In this paper, we implement a controller into our algorithm so that the rate adjustment is based on the congestion state of the system and may further satisfy stability requirements by choosing the proper parameters. This detail is, however, been omitted due to space limit.

We introduce a new algorithm termed *distributed bandwidth fair allocation* (DBFA), which is designed to achieve the RPR key performance requirements. The algorithm, which is performed at all RPR nodes in a distributed way, uses a simple proportional control mechanism to allocate the link bandwidth among all the competing flows crossing this link in a weighted manner. In order to realize global coordination on the entire RPR ring, we propose to use a certain control packet that runs around the ring to collect load information on every node. The collected information is written into the control packet in a common field which we call *fairness control field* (FCF). In our RPR dual-ring topology, we propose to have two opposite directional control packets - one on each ring. The control packet on one ring controls the data traffic on the other ring. As the control packet propagates along the ring, each node on the ring uses the information from the FCF and dynamically adjusts its sending rate and eventually reaches its fair share of the ring bandwidth. As a result, we achieve global fairness as well as high utilization, maximal spatial reuse and stability.

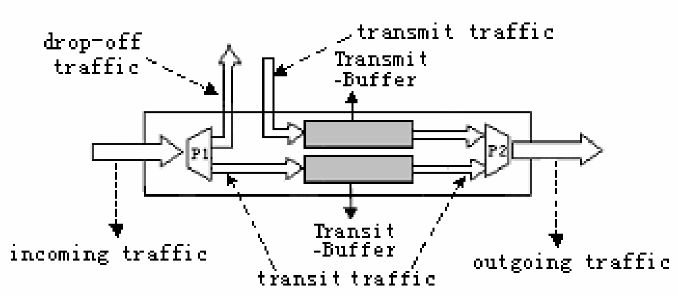

Figure 1. The data traffic process in the RPR node

control packet. Each such FCF sub-field has two functions for this node: (i) the one where the node writes its load information, and (ii) the other where the node get the control message to adjust its sending rate. Every time the control packet passes by an RPR node, the relevant FCF sub-field is updated and also the sending rate of this node is adjusted. Let us consider one of the RPR ring nodes, say node *n*, as an example, to analyze our DBFA algorithm and the operation process of which is illustrated in Figure 2.

As we can see from Figure 2, flow 1, flow 2, ⋯, flow *m* are the *m* transit flows that pass through node *n*, and flow *n* is the transmit flow originating from node *n*, note all these flows belong to the IA flow. The control packet is to collect load information and update the relevant FCF sub-field for each node and node *n* reads its own FCF sub-field when the control packet reaches it. Recall that there are two opposite directional control packets each of which is in the direction opposite to the data transmission traffic it controls. Buffer-a is used for the transmit traffic and Buffer-p is used for the transit traffic, the P-Controller is used to adjust the rate of the transmit traffic and the transit traffic according to their buffer occupancies. The bandwidth on the outgoing Link *L* of this node is shared by these flows in a weighted manner. We distinguish the transit traffic from the transmit traffic that pass through the same node and give certain priority to the former, to ensure the data transmission along the ring is successfully done. Specifically, when the occupancy of Buffer-p exceeds its target value, we adjust (reduce) the sending rates of the transit traffic through the explicit rate feedback to the source nodes, and reallocate more bandwidth on link L to the transit traffic streams simultaneously. This way, we ensure that the transit traffic streams that pass through this node without loss as well as without starving the node itself. We adopt the proportional controller to control the rate of the transit/transmit traffic according to their buffer occupancies, which can be described as follows

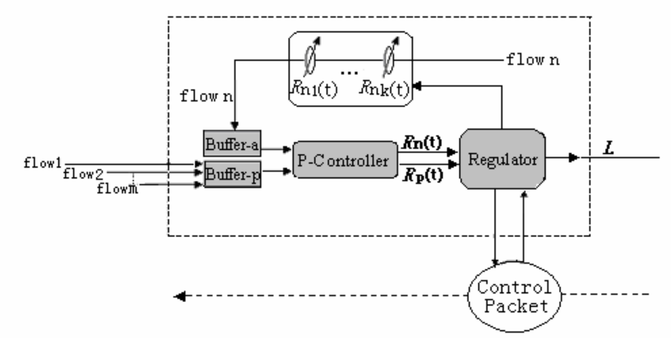

Figure 2. The operating process of the DBFA algorithm

$$R(t) = \mu - k(Q(t - \tau_b) - Q_0) \qquad (1)$$

The above formulation is used to adjust the sending rate for one flow crossing a certain link according to the maximal bandwidth *u* allocated to it and its buffer occupancy $Q(t)$ on the link, $\tau_b$ is the backward delay from the link to the source of the flow, *k* is the control parameter and $R(t)$ is the allowed sending rate for this flow.

The complete pseudocode of the DBFA algorithm is provided in Figure 3. We first design the following data structure for each IA flow on the ring

*Structure IA_flow*

```
end for
When the control packet reaching node n      // for further adjusting
for each $flow_i$                                  // update the FCF sub-filed    for $flow_i$
    $c\_flow_i.number \leftarrow max(flow_i.number, c\_flow_i.number)$;
    $c\_flow_i.rate \leftarrow min( flow_i.rate, c\_flow_i.rate)$;
end for
for node n                              //adjust node n's sending rate(per destination)
if ($flow_n .Rate<c\_flow_n .Rate$)
  $Rn_1, Rn_2, \ldots Rn_k \leftarrow flow_n .rate$;
                              //get the sending rate for each single flows within $flow_n$
else
  if ($flow_n .number>c\_flow_n .number$)
    $Rn_2\ Rn_3 \ldots Rn_k \leftarrow c\_flow_n .rate$;
                              //get sending rate for the (k-1) downstream single flows
    $Rn_1 \leftarrow U - c\_flow_n .Rate - \Sigma c\_flow_i.Rate$;
                              //get sending rate for the up single flow
  else
    $Rn_1, Rn_2, \ldots Rn_k \leftarrow c\_flow_n .rate$;
  end if
end if
End
```

Figure 3. The complete pseudocode of DBFA algorithm

ation for the transmit flow by the deflation factor *f* in order to increase the bandwidth allocated to the transit traffic streams, this is specified by the determination of the two parameters $U_a(t)$ and $U_p(t)$. On the other hand, we should adjust the sending rate of the transiting traffic streams using the explicit rate feedback to their source nodes. Under this scheme, we can ensure that the data loss of the transit traffic streams passing through the node can be prevented and in the meantime the node itself will not starve.

As we can see from the DBFA Algorithm, $flow_i$ is the local entity at node *n* to perform the bandwidth allocation for node *i* while $c\_flow_i$ is the global entity to achieve the final fair share for node *i*. Accordingly, the rate determined by proportional controller in an RPR node is only suitable for the local area, so to reach the global coordination, node *n* should further adjust its sending rate according to its own FCF sub-field in the control packet when it arrives at this node; also the FCF sub-fields in the control packet for the other relevant nodes should be updated by this node. As the control packet rotates around the RPR ring, all the nodes can drag its sending rate to the desired fair value, as a result we can achieve the global fairness for all flows and realize the high utilization on the links and the maximal spatial reuse over the entire RPR network.

## 4. SIMULATION RESULTS

In this section, we construct two simulation models to test the performance of our DBFA algorithm. These two models are as shown in Figure 4 and Figure 5. The model of Figure 4

4), flow (2, 4) and flow (3, 4) on link (3, 4) obtain their fair shares of 50 Mbps, 34 Mbps and 18 Mbps, respectively. While flow (1, 3) and flow (1, 2) obtain their fair shares of 16 Mbps and 33 Mbps, respectively. All these rate allocations lead to the full utilization of the bandwidth on each link. We have demonstrated that all RPR requirements are achieved: fairness, high utilization as well as maximal spatial reuse.

## 5. CONCLUSIONS

We have proposed a bandwidth allocation algorithm for RPR networks termed DBFA, which is able to achieve the key performance objectives of RPR networks, i.e., fairness, high utilization and spatial reuse. The algorithm operates at each RPR node in a distributed manner, and the sending rates of various flows passing through a node are controlled by using the well-established proportional control method which is known to provide stability in many cases. We use a rotating global control packet to collect and deliver the fairness control message for every node, in order to achieve the global coordination, fairness and efficiency over the entire RPR ring. Our detailed analysis and simulation results provide evidence for the satisfactory performance of the proposed scheme.

## REFERENCES

[1] RPR Alliance, “An Introduction to Resilient Packet Ring Technology,” October 2001. http://www.rpralliance.org/.

[2] C. N. Hawkins, J. Green, M. Sharma, and K. Vasani, “Resilient Packet Rings for Metro Networks,” August 2001. http://www.rpralliance.org/.

[3] F. Davik, M. Yilmaz, S. Gjessing and N. Uzun, “IEEE 802.17 resilient packet ring tutorial,” *IEEE Communications Magazine*, vol. 42, no. 3, March 2004, pp. 112-118.

[4] Andrew S. Tanenbaum, *Computer Networks* (Third Edition). Prentice Hall, Inc.1996.

[5] I. Cidon and Y. Ofek, “MetaRing – A Full-Duplex ring with Fairness and Spatial Reuse,” *IEEE/ACM Transaction on Communication*, vol. 41, no. 1, January 1993, pp. 110 -120.

[6] D. Tsiang and G. Suwala, “The Cisco SRP MAC Layer Protocol,” Internet RFC 2892, August 2000.

[7] V. Gambiroza, Y. Liu, P. Yuan, and E. Knightly, “High Performance Fair Bandwidth Allocation for Resilient Packet Rings,” in *Proceedings of the 15th ITC Specialist Seminar on Traffic Engineering and Traffic Management* ,Wurzburg, Germany, July 22-24 2002.

[8] V. Gambiroza, P. Yuan, L. Balzano, Y. Liu, S. Sheafor, and E. Knightly, “Design, analysis, and implementation of DVSR: a fair high-performance protocol for packet rings,” *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, January 2004, pp. 85-102.

[9] A. Mekkittikul, “Alladin Proposal for IEEE Standard 802.17,” Draft 1.0, November2001. http://grouper.ieee.org/groups /802/17/documents/drafts/zz_draft_04.zip.

[10] J. Kao et al, “Gandalf Proposal for IEEE Standard 802.17,” Draft 0.4, Noverrnber2001. http://grouper.ieee.org/groups /802/17/documents/drafts/gandalf_04Plus.pdf.

[11] J. Kao et al, “Darwin Proposal for IEEE Standard 802.17,” Draft 1.0, Jan.2002.http://www.ieee802.org/17/documents /presentations/jan2002/ib_fairdraft_02.pdf.

[12] E. Knightly, L. Balzano, V. Gambiroza, Y. Liu, S. Shaefor, P. Yuan, and H. Zhang, “Achieving High Performance with Darwin’s Fairness Algorithm,” IEEE 802.17 Resilient Packet Ring Standard Meeting, St. Louis, MO, March 2002.

[13] Liansheng Tan, Hanwu Wang and Moshe Zukerman, “Adapative Bandwidth Allocation for Metropolitan and Wide-Area Networks,” *IEEE Communications Letters*, vol. 9, no. 6, June 2005.

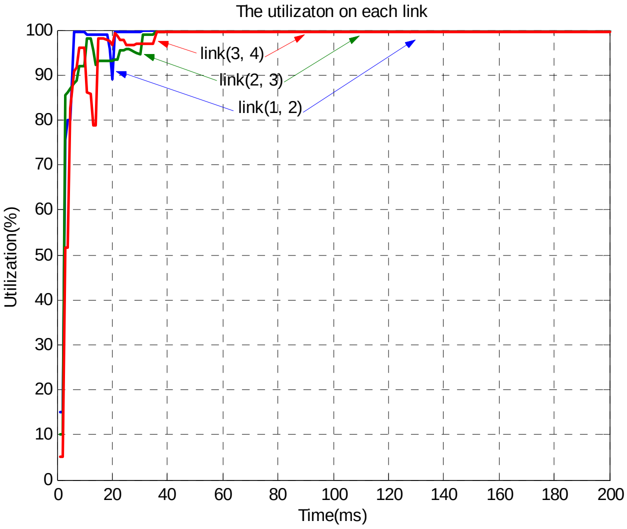

Figure 9. The utilization on the three links