RESILIENT PACKET RING: MODELLING OF FAIRNESS ALGORITHM

Bogdan Ušćumlić 1, supervisor Michel Morvan 2
1The School of Electrical Engineering in Belgrade
2Ecole nationale supérieure des télécommunications de Bretagne

I INTRODUCTION

This work describes our approach to the modelling and the simulation of the fairness algorithm of Resilient Packet Ring protocol. The simulation is written in C++ by using object-oriented simulation technique of discrete-event systems.

II OVERVIEW OF RESILIENT PACKET RING

- Definition of RPR. Resilient Packet Network (RPR) is a new technology for the metropolitan area networks (MAN), that supports the exchange of data between the nodes relied in a network composed of two rings. Recently, the protocol RPR was officially published as IEEE Standard 802.17.

Simply said, RPR protocol defines the MAC layer for Ethernet networks by offering:
1) Spatial reuse
2) Fast and efficient protection (like in SDH)
3) Fair and efficient utilisation of the bandwidth

Spatial reuse is defined as a simultaneous transfer of the independent traffic on the non-overlapped segments of the ring.
The protection is fast and efficient. The stations use one of two protection mechanisms defined by the protocol: Wrap or Steer.

Fast and efficient utilisation of the available bandwidth on the ring is provided thanks to the fairness algorithm of protocol RPR.

- Ring definition. RPR employs the ring structure composed of small rings called ringlets, which are unidirectional and counter-rotating. Each ringlet is composed of links with data flows in the same directions. The ringlets are marked as ringlet0 and ringlet1, as it is presented on the Figure 1. This association of the links to the each of the ringlets doesn’t change in the case of change in state of links or stations.

Figure 1. – The ring structure

All links work at the same rates, but the delays provided by them could be different.

The portion of ring separated by two neighbouring stations is called span (figure 1). Each span is composed of two links, which are unidirectional but transmit in opposite directions.

- RPR layer model. Figure 2 presents the layer model of RPR protocol and its comparison to the OSI (Open Systems Interconnect) reference model.

The protocol defines next sublayers:
- MAC control
- MAC datapath and
- reconciliation sublayer.

These sublayers support MAC service interface and PHY service interface. The MAC clients use the service primitives to communicate with other clients of MAC via MAC service interface. MAC service interface is also used for the transport of local control information between MAC and the MAC clients. The sublayer MAC control controls MAC datapath, saves the state of MAC and the coordination with the MAC control sublayer of the other MACs and controls the transfer of data between the MAC and its client.

The sublayer MAC datapath provides the functions for data transport for each of the ringlets. The MAC employs PHY service interface for the transmission and the receiving of frames via the physical medium.

- Spatial reuse. Spatial reuse is the bandwidth reuse capability of the procedures manipulating with the packets on the ring. These characteristics of RPR protocol could be formally defined as the “simultaneous transfer of the independent traffic on the non-overlapped portions of the ringlet”.

Traditional data rings as Token Ring or FDDI, for instance, use the source stripping of the sent packets and the token procedures to control the access to the ring. The packets used to circulate all around the ring before they have been stripped out by the source station.

On the contrary, RPR provides destination striping of the unicast packets. As the nodes are now able to transmit the packets independently, without waiting for a token, there are the considerable possibilities for the exploitations of the
bandwidth on the other portions of the ring, as it is illustrated on the Figure 3.

Figure 3: Spatial reuse

On the other side, multicast packets will be deleted at the source station.

- **Classes of traffic and frame types.** The RPR protocol defines different classes of traffic. The fairness algorithm is previewed for some of these classes. It means that traffic could be fairness eligible (FE) or not, as we will more explain now.

The traffic classes are A, B or C (table 1). Class A has the highest priority, and class C the lowest. For the class A service, RPR provides the transport of traffic with the low level of jitter. Because of that, the transfer delay of that class of traffic is also limited. The maximum value of rate is limited for the class A. All the traffic with a rate that is greater of this limited value will be rejected. There are two traffic subclasses of the class A. They are: subclassA0 and subclassA1.

<table>
<thead>
<tr>
<th>Class/Subclass</th>
<th>Bandwidth guarantees</th>
<th>Service type</th>
<th>Jitter</th>
<th>Fairness eligible</th>
</tr>
</thead>
<tbody>
<tr>
<td>A/subclassA0</td>
<td>yes</td>
<td>attributed</td>
<td>low</td>
<td>no</td>
</tr>
<tr>
<td>A/subclassA1</td>
<td>yes</td>
<td>attributed</td>
<td>low</td>
<td>no</td>
</tr>
<tr>
<td>B/CIR</td>
<td>yes</td>
<td>attributed</td>
<td>limited</td>
<td>yes</td>
</tr>
<tr>
<td>B/EIR</td>
<td>no</td>
<td>opportunistic</td>
<td>unlimited</td>
<td>yes</td>
</tr>
<tr>
<td>C/</td>
<td>no</td>
<td>opportunistic</td>
<td>unlimited</td>
<td>yes</td>
</tr>
</tbody>
</table>

Table 1: Traffic classes

Traffic of class B could be further separated to the traffic of class B – CIR and B – EIR. These classes of traffic are different because their rates are different. For the traffic of class B at rate that is less or equal as the CIR (committed information rate) value, RPR provides a limited delay and this class of traffic is called class B – CIR. For the traffic of class B, with higher rates only, best-effort transfer of data is provided and this class of traffic is called class B – EIR. For the class C traffic, only best-effort transport of data is provided.

The bandwidth is previewed (attributed) for the traffic of class A and class B – CIR. The bandwidth attributed to traffic of subclassA0 is not reclaimable by other lower-priority traffic classes. The bandwidth attributed to subclassA1 and B – EIR is reclaimable by other lower-priority classes.

There are four different frame types defined by the protocol:

1) data frame (transport of the traffic)
2) control frame (network managing)
3) fairness frame (fairRate communication)
4) idle frame (synchronisation in the network)

- **Station structure.** MAC client gives the source address and the destination address to MAC. By using this information, MAC will decide which ringlet will be selected for the transport and it will send data.

MAC client gives the packet to the MAC control layer. The MAC control layer manage with MAC datapath layer. MAC client layer is not defined by the Resilient Packet Ring protocol. The station is composed of four entities (figure 4). Those entities are: MAC client, MAC and two PHY entities. The MAC entity contains one MAC control entity, a ringlet selection entity, and two datapath entities (one datapath is associated with each ringlet).

- **The packets’ paths through a station.** The simplified structure of a station using RPR is presented on the figure 5.

This time, the figure shows the paths, in a station, of the packets originating from the station itself and the neighbouring station, which have to be sent downstream. The structure presented here is the same for each ringlet.

On this figure, only the waiting queues and the most important functions are presented. Let us look at the traffic that arrives from the MAC client. MAC client will classify the data traffic into three classes: A, B or C. Useful data information originating from the MAC client will arrive to the MAC via the primitives exchanged between these two entities.

On this figure, we can see the sequence of queues and servers crossed by the packets before leaving the station. The servers from this figure, actually, represent, in a symbolic way, the functionalities of the protocol.
There are two queues previewed for the packets arriving from the upstream station. The queues are PTQ (Primary Transit Queue) and STQ (Secondary Transit Queue). If a station has only PTQ, it works in single_queue mode. If a station has both PTQ and STQ, we will say that it works in dual_queue mode.

PTQ is previewed for the frames of classes A and B – CIR. STQ is reserved for the class B – EIR and class C frames. All the others queues in the system are also of this queue type, except the other two queues presented on the figure 5: Q_TX_SS and Q_TX_STAGE.

Q_TX_SS (Logical Selection Queue) is a queue where the principle of packet selection depends also of packet’s characteristics and system’s state and not only of the packet’s position in the queue.

Q_TX_STAGE is also an interesting queue. It is previewed for only one data frame in the same time. A frame in STAGE queue will always be provided by the StageQueueSelection server. As soon as Q_TX_STAGE queue becomes empty, SQS server will choose one of the frames from Q_TX_SS queue, by looking after the actual system state and the characteristics of frames in the Q_TX_SS queue.

![Figure 6: A shaper model](image)

The DualQueueTransmit server will choose a frame for the transport from one of the queues Q_TX_STAGE, Q_TX_PTQ or Q_TX_STQ in a way determinated by scheduling_algorithm.

All the traffic entering in the station and originating from the station itself will be first affected by the shapers of traffic. What the shapers are doing?

We could imagine a shaper like a token bucket that is periodically filled by certain number of tokens (figure 6). In the same time, the shaper will lose some tokens with each packet passing by, depending of the size of the packet.

By using the shapers realized in described way, the maximal traffic rate could be limited (determinated). Because of that, the shapers for the classes A and B – CIR are realized as it is described.

The shapers function together with the fairness algorithm. This algorithm is the core of the RPR protocol. Thanks to this protocol, RPR is capable to dynamically change the rates of the traffic on a link (on the links). The traffic managed by the fairness algorithm is only the class C traffic, or the traffic of the class B – EIR, ie. the traffic that is fairness eligible. An instance of the fairness algorithm’s functions exists for each ringlet.

II FAIRNESS ALGORITHM

• Basic characteristics. The fairness algorithm is a distributed and dynamic algorithm which has a lot of functions. Its goal is to provide, in the same time, a fair and efficient use of the bandwidth and spatial reuse. The main features of the fairness algorithm are:
  1) The fairness algorithm operations are performed independently for each ringlet.
  2) The fairness algorithm’s control information are transported by the ringlet that is opposite to the ringlet used for the transport of the observed flow.
  3) The algorithm regulates only the class B – EIR and class C traffic (the traffic that is fairness eligible).
  4) The fair rates computation is associated with a source station.
  5) Separated regulation is provided for all the traffic that is fairness eligible and is added to the ring and for the part of the added traffic that passes the point of congestion.
  6) The calculated fair rates are proportional to the administrative weights of the stations.
  7) The ringlet capacity that is not explicitly attributed is treated to be available capacity.
  8) The ringlet capacity attributed to the classes A and B – CIR, but not in use, is treated to be available capacity.
  9) The algorithm is previewed for both working modes: single_queue and dual_queue mode.
  10) The algorithm supports two managing methods of the traffic: conservative mode and aggressive mode.

• The parking lot scenario. As an illustration of the way in which the fairness algorithm uses the available ring bandwidth, we have chosen the example presented on the figure 7.

![Figure 7: Parking lot scenario](image)

The example shows four flows sharing the link 4, between the station 4 and 5. This example is called parking lot scenario. For a fair and the most efficient use of the links’ capacity, the fairness algorithm will provide the rate of 1/4 link capacity to each of the flows (1, 5), (2, 5), (3, 5) and (4, 5). We have proved that our simulation behaves well in the cases when 2, 3 and 4 flows simultaneously use (share) the same link.

• Congestion control. We will now define the principal value calculated by the fairness algorithm. That value is called fairRate, but first let’s analyse a congestion example on the figure 8. A station is congested if there are too much traffic, which is determined by congestion conditions.
These conditions depend on the station’s working mode (single_queue or dual_queue).

At the beginning of the simulation only flow 2 is active. After some period of time, flow 1 is also activated. The fairness algorithm should limit the maximal rate value of the flows, because the available links’ capacity is not big enough to support both flows in the same time. As we can see from the figure 9, where the rate of the flow 1 is presented, the fairness algorithm has allocated the half of the link capacity to each of the flows 1 and 2. Obviously, the obtained values converge to the values that are theoretically predicted.

### Insight

The fairness algorithm is designed to ensure a fair distribution of bandwidth among the users. In the context of the Resilient Packet Ring (RPR) protocol, the algorithm is crucial for managing traffic in a way that avoids congestion and maximizes the utilization of the network resources.

### Implementation

The fairness algorithm was implemented using C++ and an object-oriented simulation technique of discrete-event systems. The results obtained from the simulation proved that the model works fine in the scenario of a parking lot with two different flows of the lowest priority traffic that share the same link.

### Future Work

The final goal of this project is to compare the RPR solution with the Synchronous Digital Hierarchy (SDH) solution. This comparison will provide insights into the performance and efficiency of each technology in real-world scenarios.

---

**LITERATURE**


**Acknowledgments:** I would like to thank my supervisor Mr. Michel Morvan from the school “Ecole nationale supérieure des télécommunications de Bretagne”.

**Abstract:** The simulation of the fairness algorithm of Resilient Packet Ring protocol is written in C++ by using object-oriented simulation technique of discrete-event systems. The results obtained by our programme and shown here prove that our model works fine in a case of parking lot scenario with two different flows of the lowest priority traffic that share the same link.