An Effective Fair Bandwidth Allocation in Resilient Packet Ring

Wen-Shiang Tang, Wei-Chien Wang, Chung-Ju Chang* and Yao-Ting Hsieh
Department of Electrical Engineering
National Chiao Tung University
Hsinchu 300, Taiwan
E-mail: cjchang@mail.nctu.edu.tw

Abstract— **In this paper, we propose an effective fuzzy local fair-Rate generator (FLFG) for resilient packet ring (RPR) to achieve better utilization fairness and convergence time. The FLFG is composed of three components: adaptive fairRate calculator (AFC), fuzzy congestion detector (FCD), and fuzzy fairRate generator (FFG). AFC produces an estimated fairRate in the meantime FCD indicates the congestion degree of station. Finally, FFG adopts the two outputs from AFC and FCD to generate precise local fairRate. Simulation results show that FLFG performs a stable fairness algorithm and improves at least about by 200% in converge time of the fairRate over the *aggressive mode* (AM) and the distributed bandwidth allocation (DBA) algorithms.**

I. INTRODUCTION

The resilient packet ring (RPR) is a ring based network for high-speed metropolitan area networks (MANs) and is constructed by several pairs of two unidirectional links between stations [1]. The RPR can get rid of deficiencies of some MANs, such as SONET and high-speed Ethernet, according to some noticeable properties like spatial reuse and new fairness mechanisms of bandwidth allocation.

The spatial reuse allows a frame to be removed from the ring at its destination so that the bandwidth on next links can be re-used at the same time. Also, the fair bandwidth allocation avoids stations at upstream transmitting too many low-priority frames to cause stations at downstream system congestion. RPR needs congestion control to enhance the fair bandwidth division in the congestion domain which is defined in the IEEE 802.17 [2], [3]. The congestion control implemented in each station should periodically generate an *advertised fairRate* to advertise its upstream station for regulating the added fairness eligible (FE) traffic flow defined in IEEE 802.17 [2], [3]. The advertised fairRate should be determined referring to the local fairRate, the received fairRate, and the congestion degree of the station. The local fairRate is generated by a fairness algorithm, and the received fairRate is the advertised fairRate from the downstream station.

Two key factors affect performance of the fair bandwidth allocation: congestion detection and fairness algorithm. If the congestion detection is too rough, it would lower the network's throughput or raise frame loss. The fairness algorithm should consider the most important performance issues of FE traffic flows: stability, fairness, convergence time, and throughput loss caused by the FE traffic flow oscillation. The stability would avoid the oscillation of regulated FE traffic flows, which would cause the throughput loss. If a fairness algorithm referees a "ring ingress aggregated with spatial reuse (RIAS)" fairness, it has been proved that the algorithm will achieve high system utilization [4]. It is because the RIAS has two key properties. The first property is that an *ingress-aggregated* (IA) flow fairly sharing the bandwidth on each link, relating to other IA flows on the same link, where an IA flow is the aggregate of all flows originating from a given ingress station. The second property is that the maximal spatial reuse subjecting to the first property. Thus, the bandwidth can be reclaimed by IA flows when it is unused. In summary, the RIAS is a max-min fairness with traffic granularity of IA flow. The convergence time is the time interval between the instant of starting the congestion occur and the instant that the amount of arriving specified traffic flow approaches the ideal fairRate which meets the the RIAS fairness. Therefore, a fairness algorithm should achieve not only high stability based on the RIAS fairness but also low convergence time and flow oscillation.

The aggressive mode (AM) fairness algorithm has been proposed in IEEE 802.17. It would suffer from severe oscillations and bandwidth utilization degradation [2]-[6]. It is because AM issues a un-limited fairRate, called FullRate, as its advertised fairRate when the station is released from congestion. Several fairness algorithms were proposed to solve this problem and some of them were designed based on the RIAS fairness [4], [5]. Gambiroza et al. proposed a distributed virtual-time scheduling in rings (DVSR) [4]. Unfortunately, it is at the expense of a high computational complexity $O(N \log N)$, where N is the number of stations in the ring. Alharbi and Ansari proposed a distributed bandwidth allocation (DBA) fairness algorithm with a low computational complexity $O(1)$ [6], [7]. However, whenever the effect of propagation delay is severe, the DBA would not be a stable local fairRate algorithm. It is due to the face that the amount of the arriving transit FE traffic flows which is referred by DBA to generate the local fairRatebut only is measured during a short frame time. This short-term amount is easily influenced by the effect of the propagation delay, which starts from a station sending its advertised fairRate and ends the corresponding transit traffic flows arriving the station. If the propagation delay is large, the short-term arriving transit FE traffic flows would be largely varied and makes the generation of local fairRate unstable (incorrect).

In this paper, we propose an effective local fairRate generator based on fuzzy logic theory [9] and moving average technique. The effective local fairRate generator, named fuzzy local fairRate generator (FLFG), can meet the RIAS fairness and reflect timely the congestion status of station. The FLFG is sophisticatedly configured into three functional blocks: adaptive fairRate calculator (AFC), fuzzy congestion detector (FCD), and fuzzy fairRate generator (FFG). It first pre-produces a local fairRate to meet the RIAS fairness and diminish the effect of propagation delay by AFC. Also, the FLFG evaluates the congestion degree of a station, denoting the forwarding capacity of added FE traffic flows at the station and buffering capacity of the STQ, by FCD. Finally, the FLFG generates a precise local fairRate by FFG. The

CHINACOM 2010, August 25-27, Beijing, China

DOI 10.4108/chinacom.2010.93

FFG finely adjusts the pre-produced local fairRate from AFC according to the congestion degree of the station from FCD, using fuzzy logics based upon domain knowledge. Simulation results show that the FLFG has better performance than AM and DBA in a large parking lot scenario with both greedy traffic flows and various finite traffic flows. Unfortunately, DBA cannot stabilize all flows in the two scenarios.

II. RPR OVERVIEW

A. Ring and Station Structures

Assume that a resilient packet ring (RPR) with N stations is constructed by two unidirectional, counter-rotating ringlets, named ringlet-0 and ringlet-1. Each station has two pairs of input and output ports to communicate with neighbor stations. Station X (Y) is said to be a upstream (downstream) node of station Y (X) on ringlet-0 or ringlet-1 if the station Y (X) traffic becomes the received traffic of station X (Y) on the referenced ringlet. There are three classes of service for RPR. The classA is used for real-time services and it has subclassA0 for reserved bandwidth and subclassA1 for reclaimable bandwidth. The classB is targeted for near real-time services, and it also has two subclasses: classB-CIR (committed information rate) which requires the bounded delay and guaranteed bandwidth, and classB-EIR (excess information rate) which does not guarantee bandwidth or delay bound. The classC is intended for best effort services and has the lowest priority. Each station only reserves bandwidth for subclassA0, and the remaining bandwidth is provided for other traffic classes according to the order of subclassA1, classB-CIR, classB-EIR, and classC. The latter two low priority traffics are called the *fairness eligible* (FE) traffic and are controlled by a fairness algorithm [1]-[3].

Fig. 1 shows the station structure for ringlet-0 transmisson, which contains an ingress queue with ClassA, ClassB, and ClassC queues, a transit queue with primary transit queue (PTQ) and secondary transit queue (STQ), a scheduler, the fuzzy local fairRate generator (FLFG), and a fairness control unit. The ClassX queue, X = A, B, or C, stores the added classX traffic to the station. The PTQ (STQ) stores the transiting classA and classB-CIR (classB-EIR and classC) frames. The scheduler decides the transmitting order. If the STQ occupancy is less than the *stqHighthreshold* defined in the IEEE802.17 [1], the order is PTQ, ClassA, ClassB, ClassC, and STQ; otherwise, it is PTQ, ClassA, ClassB, STQ, and ClassC. The FLAG generates a local fairRate at every time nT, denoted by $f_l(n)$, where n is a positive integer and T is the duration of an agingInterval. Notice that f_l is also generated per agingInterval in DBA but is generated only when the station is in congestion in AM. The fairness control unit usually refers to both $f_l(n)$ and the received fairRate, denoted by $f_r(n)$, to determine an advertised fairRate, denoted by $f_v(n)$, and then sends $f_v(n)$ to upstream stations to regulate traffic flows, at every agingInterval time nT.

The advertised fairRate generated by the fairness control unit are described as follows. The f_v would be set to f_l if f_r is smaller than f_l and larger than the bandwidth rate of the transit FE traffic flows which will pass through the originally congested station. Otherwise, it is set to be min(f_l, f_r). Here we also describe the advertised fairRate generated by AM below. When the station is congestion free, the f_v is set to be the FullRate if the f_r is larger than the bandwidth rate of the transit FE traffic flows which will pass through the originally congested station; to be f_r, otherwise. The FullRate is a specially advertised fairRate to indicate that the station does not need to limit its added FE traffic flow. When the station is in congestion, the f_v is set to be f_l if the f_r is FullRate; to be min(f_l, f_r), otherwise. Note that the congestion is occurred at a station for AM if the STQ occupancy of the station is larger than the *stqLowthreshold*, defined in IEEE802.17 [1]. Also, the originally congested station is known to the observation station since the message of the advertised fairRate contains a field to record it [1]; the f_l is the added FE traffic flow rate to the network.

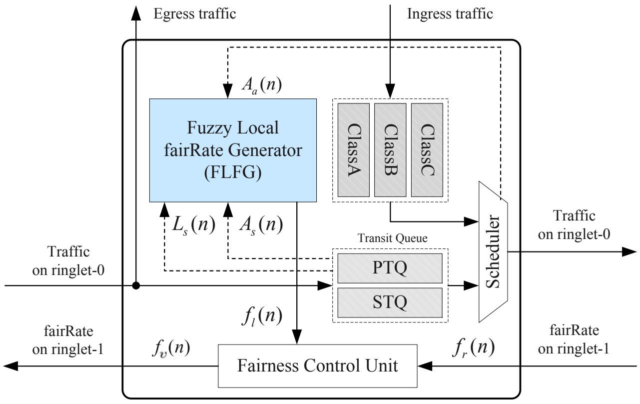


Fig. 1. RPR station structure

III. FUZZY LOCAL FAIRRATE GENERATOR

The proposed fuzzy local fairRate generator (FLFG), shown in Fig. 2, is composed of an adaptive fairRate calculator (AFC), a fuzzy congestion detection (FCD), and a fuzzy fairRate generator (FFG). During the nth agingInterval which is from time $(n-1)T$ to time nT, the FLFG determines $f_l(n)$ by referring to the arriving FE traffic flows to STQ, denoted as $A_s(n)$, the added FE traffic flow to the network, denoted as $A_a(n)$, and STQ occupancy, denoted as $L_s(n)$. The AFC pre-generates a local fairRate, called p-fairRate and denoted by $f_p(n)$, which satisfies the RIAS fairness. Its design imitates the DBA's generation of local fairRate, but it would overcome the unstable (incorrect) local fairRate generation by DBA when the propagation delay is significant. Instead of using the short-term arriving transit FE traffic flows, it calculates a proper average of the arriving transit FE traffic flows by *moving average technique* to mitigate the effect of the propagation delay. The FCD appraises the congestion status of station using fuzzy logics. Its design can softly detect the congestion degree of the station in each agingInterval n, denoted by $D_c(n)$, considering not only the STQ occupancy but also the amount of the arriving transit FE traffic flows at the queue. The latter term denotes the change rate of the STQ occupancy which would play an important role in the congestion detection. Finally, the FFG generates a precise local fairRate by fine-tuning the p-fairRate from AFC, referring to the congestion degree from FCD, and further using domain knowledge designed by fuzzy logics. The FLAG would avoid serious regulating FE traffic flows to decrease the throughput or excessive relaxing the traffic flows to increase the frame losses.

A. Fuzzy Congestion Detector (FCD)

The FCD measures arrival rate to STQ, denoted by $A_s(n)$, and occupancy of STQ, denoted by $L_s(n)$, to estimate congestion condition. After measuring congestion degree, it provides diversities of congestion control under different congestion degree to have a more precise decision. As shown in Fig. 2, $A_s(n)$ and $L_s(n)$ are the inputs of FCD and the output of FCD is a numerical value $D_c(n)$ between 0 and 1, which indicates congestion

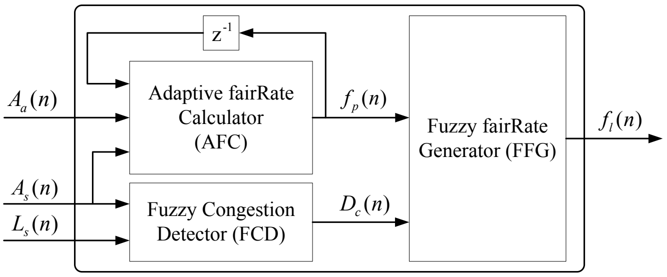


Fig. 2. FLFG structure

degree. We define the term set for $A_s(n)$ as $T(A_s(n)) = \{$Low (L), Medium (M), High (H)$\}$; for $L_s(n)$ as $T(L_s(n)) = \{$Short (S), Long (L)$\}$; for $D_c(n)$ as $T(D_c(n)) = \{$Very Low (VL), Low (L), Medium (M), High (H), Very High (VH)$\}$. Here, the triangular function $f(x; x_0, a_0, a_1)$ and the trapezoidal function $g(x; x_0, x_1, a_0, a_1)$ are used to define the membership functions for terms in the term set. These two functions are given by

$$f(x; x_0, a_0, a_1) = \begin{cases} \frac{x-x_0}{a_0} + 1, & \text{for } x_0 - a_0 < x \le x_0, \\ \frac{x_0-x}{a_1} + 1, & \text{for } x_0 < x < x_0 + a_1, \\ 0, & \text{otherwise,} \end{cases}$$

and

$$g(x; x_0, x_1, a_0, a_1) = \begin{cases} \frac{x-x_0}{a_0} + 1, & \text{for } x_0 - a_0 < x \le x_0, \\ 1, & \text{for } x_0 < x \le x_1, \\ \frac{x_1-x}{a_1} + 1, & \text{for } x_1 < x < x_1 + a_1, \\ 0, & \text{otherwise,} \end{cases}$$

where x_0 in $f(\cdot)$ is the center of the triangular function; $x_0(x_1)$ in $g(\cdot)$ is the left (right) edge of the trapezoidal function; $a_0(a_1)$ is the left (right) width of the triangular or the trapezoidal function. The corresponding membership functions of S and L in $T(L_s(n))$ are denoted by $\mu_S(L_s(n)) = g(L_s(n); 0, 0.125Q, 0, 0.25Q)$ and $\mu_L(L_s(n)) = g(L_s(n); 0.35Q, Q, 0.25Q, 0)$, where Q is STQ size. Note that the 0.125 times STQ size is set as the *stqLowthreshold* to judge the light congestion degree, and the 0.25 times STQ size as the *stqHighthreshold* to judge the heavy congestion degree in IEEE 802.17 [1]. The corresponding membership functions of L, M, and H in $T(A_s(n))$ are denoted by $\mu_L(A_s(n)) = g(L_s(n); 0,\ 0.125U, 0, 0.375U)$, $\mu_M(A_s(n)) = f(A_s(n); 0.5U, 0.25U, 0.25U)$, and $\mu_H(A_s(n)) = g(A_s(n); 0.875U, U, 0.375U, 0)$, respectively, where U represents unreserved rate. For the reason of simplicity in computation of defuzzification, let the membership functions for VL, L, M, H, VH in $T(D_c(n))$ be all fuzzy singletons and they are defined as $\mu_{VL}(D_c(n)) = f(D_c(n); 0, 0, 0)$, $\mu_L(D_c(n)) = f(D_c(n); 0.25, 0, 0)$, $\mu_M(D_c(n)) = f(D_c(n); 0.5, 0, 0)$, $\mu_H(D_c(n)) = f(D_c(n); 0.75, 0, 0)$, and $\mu_{VH}(D_c(n)) = f(D_c(n); 1, 0, 0)$. The membership functions of $T(D_c(n))$ distribute uniformly between 0 and 1.

TABLE I
THE RULE BASE OF FCD

Rule	$L_s(n)$	$A_s(n)$	$D_c(n)$	**Rule**	$L_s(n)$	$A_s(n)$	$D_c(n)$
1	S	L	VL	2	S	M	VL
3	S	H	L	4	L	L	M
5	L	M	H	6	L	H	VH

As shown in Table I, there are 6 fuzzy rules for FCD. As shown in Table I, the order of significance of the input linguistic variables is $L_s(n)$ then $A_s(n)$. The station with high occupancy of STQ would be in high congestion degree, and it would be in higher (medium) congestion degree if the arriving FE traffic flows to STQ is also high (low). The fuzzy congestion detector adopts the max-min inference method for inference engine because it is suitable for real-time operation. To explain max-min inference method, we take rule 1 and rule 2, which have the same control action "$D_c(n)$ is VL", into consideration. Applying the "min" operator, we obtain the membership function values of the control action "$D_c(n)$ is VL" of rule 1 and rule 2, denoted by $m_1(n)$ and $m_2(n)$, by

$$m_1(n) = \min\{\mu_S(L_s(n)),\ \mu_L(A_s(n))\},$$

$$m_2(n) = \min\{\mu_S(L_s(n)),\ \mu_M(A_s(n))\}.$$

Subsequently, applying the "max" operator yields the overall membership value of the control action "$D_c(n)$ is VL", denoted by $w_{VL}(n)$, by

$$w_{VL}(n) = \max\{m_1(n),\ m_2(n)\}.$$

The overall membership functions of the control action L, M, H, and VH for $D_c(n)$, denoted by $w_L(n)$, $w_M(n)$, $w_H(n)$, and $w_{VH}(n)$, respectively, can be obtained in a similar way. After inferring all rules, the FCD uses defuzzification of center of area (COA) for defuzzifier. The output value $D_c(n)$ is obtained as follows:

$$D_c(n) = \frac{\sum_{i \in \{VL,L,M,H,VH\}} d_i \times w_i(n)}{\sum_{i \in \{VL,L,M,H,VH\}} \times w_i(n)},$$

where, $d_{VL} = 0$, $d_L = 0.25$, $d_M = 0.5$, $d_H = 0.75$, and $d_{VH} = 1$. Therefore, $D_c(n)$ indicates the congestion degree with a crisp value.

B. Adaptive fairRate Calculator (AFC)

The adaptive fairRate calculator (AFC) adopts the moving average technique on the short-term arriving FE traffic flows, trying to mitigate the effect of propagation delay on the generation of local fairRate by the DBA [6]. During the n-th agingInterval, the AFC first takes the *moving average* of arriving transit FE traffic flows to STQ, $A_s(n)$. Denote the average by $\tilde{A}_s(n)$ and give it by

$$\tilde{A}_s(n) = \Sigma_{i=n-k+1}^{n} A_s(i)/k, \tag{1}$$

where k is the size of observation window. The k is the sum of two kinds of the data frame trip time: one is the time from the furthest source to this observation station, and the other is the time from this station to originally congested station. It is because the FE traffic flow of a station in this interval would be regulated by an advertised fairRate which is sent out from one of the stations in the interval. The $\tilde{A}_s(n)$ will not vary too much and become more stable.

Then the AFC computes the effective number of IA flows during the n-th agingInterval, denoted by $M(n)$, which is obtained by

$$M(n) = \frac{\tilde{A}_s(n) + A_a(n)}{f_p(n-1)}. \tag{2}$$

The AFC fairly allocates the remaining bandwidth to these effective IA flows, which would be $\frac{1}{M(n)}(C - (A_s(n) + A_a(n)))$. Finally, the AFC calculates the $f_p(n)$ by adding up the previous p-fairRate, $f_p(n-1)$, and the fairly shared bandwidth. The $f_p(n)$ is given by

$$f_p(n) = \text{mim}\left\{C,\ f_p(n-1) + \frac{1}{M(n)}\Big[C - (A_s(n) + A_a(n))\Big]\right\}, \tag{3}$$

where C is the unreserved bandwidth for FE traffic flows per agingInterval used to denote the upper bound of the local fairRate.

C. Fuzzy fairRate Generator (FFG)

The FFG refers the *p*-fairRate, $f_p(n)$, and the congestion degree, $D_c(n)$, as the input variables to generate a proper and robust local fairRate, $f_l(n)$. The local fairRate $f_l(n)$ affects both the fairness performance and the bandwidth utilization. Define the term set with six terms for $f_p(n)$ as $T(f_p(n)) =\{$Extremely Low (EL), Pretty Low (PL), Slightly Low (SL), Slightly High (SH), Pretty High (PH), Extremely High (EH)$\}$; the term set with three terms for $D_c(n)$ as $T(D_c(n)) =\{$Low (L), Medium (M), High (H)$\}$; and the term set with eleven terms for $f_l(n)$ as $T(f_l(n)) =\{$Extremely Low (EL), Very Low (VL), Pretty Low (PL), Low (L), Slightly Low (SL), Medium (M), Slightly High (SH), High (H), Pretty High (PH), Very High (VH), Extremely High (EH)$\}$. Note that the number of the terms in $T(f_l(n))$ would be larger than that of $T(f_p(n))$ for better performance.

The membership functions for terms $EL, PL, SL, SH, PH,$ and EH in $T(f_p(n))$ are defined as $\mu_{EL}(f_p(n)) = f(f_p(n); 0, 0, 0.3C)$, $\mu_{PL}(f_p(n)) = f(f_p(n); 0.2C, 0.2C, 0.2C)$, $\mu_{SL}(f_p(n)) = f(f_p(n); 0.4C, 0.2C, 0.2C)$, $\mu_{SH}(f_p(n)) = f(f_p(n); 0.6C, 0.2C, 0.2C)$, $\mu_{PH}(f_p(n)) = f(f_p(n); 0.8C, 0.2C, 0.2C)$, and $\mu_{EH}(f_p(n)) = f(f_p(n); C, 0.3C, 0)$, respectively. The membership functions for terms $L, M,$ and H in $T(D_c(n))$ are defined as $\mu_L(D_c(n)) = g(D_c(n); 0, 0.125, 0, 0.375)$, $\mu_M(D_c(n)) = f(D_c(n); 0.5, 0.25, 0.25)$, $\mu_H D_c(n)) = g(D_c(n); 0.875, 1, 0.375, 0)$, respectively. The membership functions for terms in $T(f_l(n))$ are defined as fuzzy singletons, denoted by $\mu_T(f_l(n)) = f(f_l(n); x_T, 0, 0)$, where $T = EL, VL, PL, L, SL, M, SH, H, PH, VH,$ or EH, and $x_{EL} = 0$, $x_{VL} = 0.1C$, $x_{PL} = 0.2C$, $x_L = 0.3C$, $x_{SL} = 0.4C$, $x_M = 0.5C$, $x_{SH} = 0.6C$, $x_H = 0.7C$, $x_{PH} = 0.8C$, $x_{VH} = 0.9C$, $x_{EH} = C$. Notice that the center value of the triangular membership function f of each term for $f_p(n)$ is the same as the center value of the singleton function f of the same term for $f_l(n)$, where these terms are $EL, PL, SL, SH, PH,$ and EH.

There are 18 fuzzy rules for FFG. As shown in Table II, the order of significance of the input linguistic variables is $f_p(n)$ then $D_c(n)$. These fuzzy rules are set in such a way that the generation of $f_l(n)$ mainly refers to $f_p(n)$ but slightly adjusted by $D_c(n)$ so as to achieve lower convergence time and thus higher the throughput. When $f_p(n)$ is $''EL''$ or $''PL''$, $f_l(n)$ is designed to raise two levels more than $f_p(n)$ $(EL \rightarrow PL$ or $PL \rightarrow SL)$ if $D_c(n)$ is $''L''$ and $f_l(n)$ remains unchanged if $D_c(n)$ is $''H''$. This intends to increase the throughput. When $f_p(n)$ is $''SL''$,$''SH''$, or $''PH''$, $f_l(n)$ decreases one level less than $f_p(n)$ if $D_c(n)$ is $''H''$ and $f_l(n)$ increases one level larger than $f_p(n)$ if $D_c(n)$ is $''L''$. When $f_p(n)$ is $''EH''$, $f_l(n)$ should be decreased two levels less than $f_p(n)$ $(EH \rightarrow PH)$ if $D_c(n)$ is $''H''$ and $f_l(n)$ remains unchanged if $D_c(n)$ is $''L''$. This intends to achieve RIAS fairness. Finally, the defuzzifier uses the min-max method mentioned in Section III-A to generate a crisp-valued local fairRate.

IV. SIMULATION RESULTS AND DISCUSSIONS

In the simulations, settings for the environment include 10 Gbps link capacity, 100 μs propagation delay between stations, 4 Mbytes STQ size, and 100 μs agingInterval. The value of the *stqHighthreshold* is 1 Mbytes and the value of the *stqLowthreshold* is 0.5 Mbytes. Simulations for the proposed FLFG, DBA [6], and AM [3] also conducted for performance comparison. Simulation results are recorded per agingInterval. Also, assume that the reserved bandwidth is zero, and only fairness eligible (FE) traffic flow is considered.

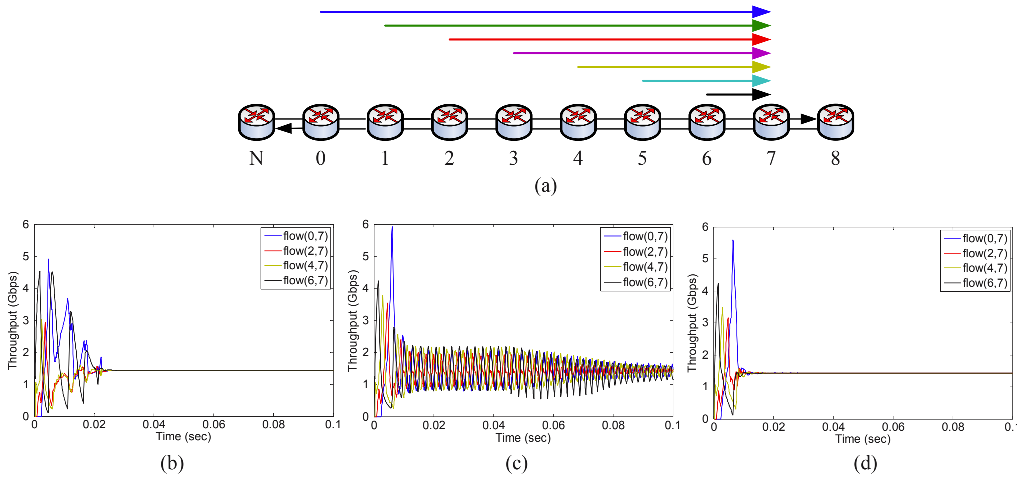
Fig. 3(a) shows a large parking lot scenario where there are containing 8 greedy stations, and Figs. 3(b), 3(c), and 3(d) present the throughput of flow(0, 7), flow(2, 7), flow(4, 7), and flow(6 ,7) at station 7 by AM, DBA, and FLAG, respectively. The propagation delay would be *large*. It can be seen that the FLFG (AM) takes 11ms (27ms) to stabilize the flows; unfortunately, DBA takes quite a long time to stabilize the traffic flows. It is because that DBA computes the number of the effective IA flows referring to both the short aggregating traffic (per agingInterval) and the pervious local fairRate to generate the current local fairRate. However, due to the large propagation delay, the correlation between the short aggregating traffic and the pervious local fairRate becomes low. Therefore, DBA cannot generate a correct local fairRate to regulate flows. Thus the flows oscillate and converge slowly. On the other hand, the FLFG can correctly generate the *p*-fairRate to meet the RIAS fairness and diminish the effect of the propagation delay to some extent. Also, the FLFG finely adjusts the *p*-fairRate to a precise local fairRate according to both the congestion degree and the effective fuzzy rules well designed by domain knowledge. In Fig. 3(a), there are more stations with greedy traffic, more aggregated traffic per agingInterval will be caused. This more aggregated traffic and the larger propagation delay would make the station congestion always occur earlier. Afterwards, the station would not have the chance to set the advertised fairRate as FullRate such that the convergence time is shorter.

Figs. 4(a), 4(b), and 4(c) present throughputs of flow(0, 7), flow(2, 7), flow(4, 7), and flow(6, 7) at station 7 by AM, DBA, and FLFG, respectively, in a large parking lot scenario, which contains 8 stations as in Fig. 3(a) but with various *finite* traffic demands. Assume that flow(0, 7) and flow(1, 7) require 2.1 Gpbs, flow(4, 7) and flow(5, 7) require 1.5 Gpbs, and flow(2, 7), flow(3, 7) and flow(6, 7) require 1.0 Gbps. It would be facts that station 6 will be the *first* one to incur congestion, and the added FE traffic flow to network at each station cannot always match its received fairRate due to the finite traffic demand at each station. Also, flow(0,7) and flow(1,7) will have the highest throughput when station 6 is in free-congestion or the remaining bandwidth is large because of their largest required traffic demands. It can be seen that at the first beginning, all flows just oscillate slightly, and then AM, and DBA oscillate all the ways, while FLFG can make all flows converge but takes 30 ms. It is because that FLFG indeed diminishes the effect of the propagation delay and generates the correct local fairRate at each agingInterval. Also, since each traffic flow is with different finite traffic demand and is much less than that of the greedy case in Fig. 3(d), the damping amplitude is smaller than that in Fig. 3(d). Moreover, the FLFG stably realizes the RIAS fairness and has higher throughput by about 2.8% than AM, and 3.5% than DBA. On the other hand, the advertised fairRate by AM is often set as FullRate in this scenario due to the 10.2 Gbps required bandwidth of the total demand traffic, slightly higher than the link capacity but much less than that of the greedy case. The aggregated traffic per agingInterval would be smaller, and the congestion, if any, could be solved by AM most of time. Thus, the flows by AM oscillate always and the flow(0,7) seriously oscillates due to its largest traffic

TABLE II

THE RULE BASE OF FFG

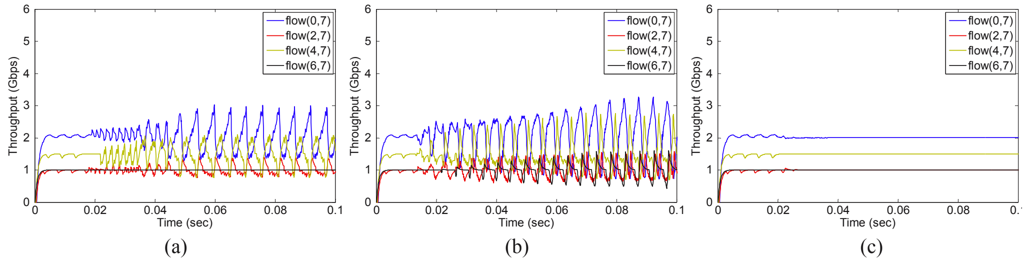
Rule	$f_p(n)$	$D_c(n)$	$f_l(n)$	**Rule**	$f_p(n)$	$D_c(n)$	$f_l(n)$	**Rule**	$f_p(n)$	$D_c(n)$	$f_l(n)$
1	EL	L	PL	7	SL	L	M	13	PH	L	VH
2	EL	M	VL	8	SL	M	SL	14	PH	M	PH
3	EL	H	EL	9	SL	H	L	15	PH	H	H
4	PL	L	SL	10	SH	L	H	16	EH	L	EH
5	PL	M	L	11	SH	M	SH	17	EH	M	VH
6	PL	H	PL	12	SH	H	M	18	EH	H	PH



(a)

(b) (c) (d)

Fig. 3. (a) Large parking lot scenario with greedy traffic, and the throughput of (b) AM, (c) DBA, and (e) FLFG.



(a) (b) (c)

Fig. 4. The throughput of (a) AM, (b) DBA, and (c) FLFG in a large parking lot scenario with various finite traffic flows.

demand. By DBA, its generation accuracy of local fairRate is susceptible to the propagation delay, as seen in Fig. 3. Also, station 0 and station 1 are the farthest ones to station 6 and flow(0,7) and flow(1,7) are with the largest traffic demand. These facts result in that flow(0,7) and flow(1,7) cannot be regulated by the station 6 quickly. This violent varying aggregation traffic per agingInterval and the effect of the propagation delay thus result in DBA generating the local fairRate improperly. Notice that if flow(0,7) requires less traffic demand, the oscillation amplitude of flows will be smaller.

V. CONCLUSIONS

In this paper, an effective fuzzy local fairRate generator (FLFG) is proposed for resilient packet ring (RPR). The FLFG can make traffic flows satisfy RIAS fairness criterion and converge to an ideal fairRate in an efficient way. Simulation results show that each flow by FLFG is indeed close to the designated rate with the smallest damping amplitude and the least convergence time, compared to conventional AM, and DBA fairness algorithms. These prove that the configuration of FLFG is indeed sophisticated, where AFC pre-generates the local fairRate using the moving average technique; FCD determines the congestion degree of station using fuzzy logics, considering not only the STQ length but also change rate of STQ length; and finally the FFG adopts the fuzzy logics and the expert's domain knowledge to precisely generate the local fairRate by fine-tuning the pre-generated local fairRate by AFC according to the congestion degree by FCD.

REFERENCES

[1] IEEE Standard 802.17, "Resilient Packet Ring (RPR) Access Method and Physical Layer Specification," 2004.

[2] F. Davik, M. Yilmaz, S. Gjessing, and N. Uzun, "IEEE 802.17 Resilient Packet Ring Tutorial," *IEEE Commun. Mag.*, vol. 42, no. 3, pp. 112 - 118, March 2004.

[3] F. Davik, A. Kvalbein, and S. Gjessing, "An Analytical Bound for Convergence of the Resilient Packet Ring Aggressive Mode Fairness Algorithm," *IEEE ICC 2005*, vol. 1, pp.281 - 287, May 2005.

[4] 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 Trans. Networking*, vol. 12, no. 1, pp. 85 - 102, February 2004.

[5] RIAS matlab codes. *http://www.ece.rice.edu/networks/RIAS/*

[6] F. Alharbi and N. Ansari, "Distributed bandwidth allocation for resilient packet ring networks," *Computer Networks*, vol.49, no. 2, pp.161 - 171, October 2005.

[7] —, "SSA: simple scheduling algorithm for resilient packet ring networks," *IEE Proceedings-Communications*, vol.153, no. 2, pp.183 - 188, April 2006.

[8] M. Yilmaz and N. Ansari, "Weighted Fairness in Resilient Packet Rings," *IEEE ICC 2007*, June 2007.

[9] C. T. Lin and C. S. G. Lee, "Neural fuzzy systems: A neuro-fuzzy synergism to intelligent systems," *Prentice Hall*, 1995.