

Efficient Bandwidth Sharing in Bus-Based Optical Access Networks

Andre-Luc Beylot², Nizar Bouabdallah¹ and Guy Pujolle¹

¹ LIP6, University of Paris 6, 8 rue du Capitaine Scott, F-75015 Paris, France
nizar.bouabdallah, guy.pujolle@lip6.fr

² ENSEEIHT - IRIT/TeSA Lab., 2 rue C. Camichel, BP7122, F-31071 Toulouse
Andre-Luc.Beylot@enseeiht.fr

Abstract. Packet-based optical access ring is becoming a promising solution in metropolitan networks. Its performance depends mainly on how optical resource sharing takes place among the different competing access nodes. In such a network fairness issues are likely to arise between upstream and downstream nodes competing to access a common data channel. This paper provides a new protocol called TCARD (Traffic Control Architecture using Remote Descriptors), based on a preventive reservation scheme, to alleviate the fairness problems. The reservation process consists in forcing the upstream nodes to ensure enough available bandwidth for downstream nodes' transmission purpose. Moreover, we develop analytical models in order to assess the access delay when TCARD is enabled. The analytical and simulation results show how the proposed solution alleviates the performance degradation and the resource sub-utilization while achieving fairness among bus nodes.

1 Introduction

Over the last decade, networks have been witnessing a perpetual growth in data traffic. This shift, driven primarily by the proliferation of internet, has created a rising demand for robust networks, with increasingly high-link capacity and node throughput. Due to the new incumbent challenges, the operators are progressively migrating toward optical core networks taking advantage of the tremendous transmission capacity offered by the optical technology. Thanks to the implementation of wavelength division multiplexing (WDM) in core networks, the relentless need for more capacity may have been satisfied. However, at the boundaries of backbone networks, especially at metropolitan and local area networks (MAN/LAN), an efficient solution for transporting and switching huge amounts of data still needs to be found.

The main goals of next-generation metropolitan networks are to provide high-speed transmissions and the integration of various services (e.g., data, voice, and video) over larger geographic areas. In this regard, three major enabling factors are identified as crucial for the evolution process of metropolitan networks' architecture: optics, packet switching, and protocol convergence.

To deal with the new requirements of next-generation metro networks, a new architecture named the dual bus optical ring network (DBORN) has been proposed.

The DBORN architecture will be described in this article, but for more detailed information the reader is invited to refer to [1]. The performance of such networks depends mainly on how optical resource sharing takes place among the different competing access nodes. Thus, both collision-free transmission and fair sharing of the common bandwidth among ring nodes must be ensured. A medium access control (MAC) protocol is needed to avoid collisions on the individual WDM channels shared among competing nodes. The proposed MAC protocol must consider the case of non-slotted WDM rings. However, even though the MAC protocol enables collision-free transmission over the shared medium, it does not address the inherent fairness issue, which is pronounced in the case of shared medium networks.

In fact, since several source nodes share a common channel, one node may grab all the available bandwidth, and possibly starve the downstream nodes competing to access the same channel. Protocols at various levels (such as MAC or CAC – Call Admission Control) must be introduced to ensure good utilization of the transmission resources and to alleviate fairness problems. In general, fairness control mechanisms limit the transmission of upstream nodes in an attempt to keep enough bandwidth for downstream stations [2]. Although these schemes may be efficient in the case of slotted WDM rings, yet, they do not perform well in the case of asynchronous transmission based architectures like DBORN. We presented in [3] analytical models and simulation results that illustrate the aforementioned issue. Indeed, we noticed that sharing the common bandwidth fairly but arbitrarily among bus nodes, like in slotted WDM ring, doesn't resolve the fairness problem within asynchronous system. In this regard, we demonstrated in [4] the inherent limitations of the token bucket access rate-based algorithm once applied to asynchronous transmission bus-based networks.

As a result, we suggest in this paper a novel control traffic protocol that aims to solve the fairness issue. The proposed solution, the so-called TCARD (Traffic Control Architecture using Remote Descriptors) and presented briefly in [4], is based on a preventive mechanism when granting access to the resource, i.e. free bandwidth is reserved by each node according to the traffic requirements of its downstream partners. In this context, analytical models are developed in an attempt to provide explicit formulas that express the mean access delay of each node of the bus-based optical access network when TCARD is considered.

The remaining parts of this paper are organized as follows. Section 2 emphasizes the MAC context including a description of the ring network along with its main features. Then, section 3 provides a detailed description of the proposed control mechanism TCARD. Analytical models for evaluating the access delay performance of each ring node, when TCARD is considered, are developed in section 4. Then, section 5 validates the accuracy of the proposed models by comparing the analytical results with that obtained by means of simulations. Furthermore we demonstrate how TCARD can achieve fairness and improves the network performances. Finally, conclusions are drawn in section 6.

2 Network Architecture and MAC Design

This section describes the DBORN architecture and the proposed MAC protocol.

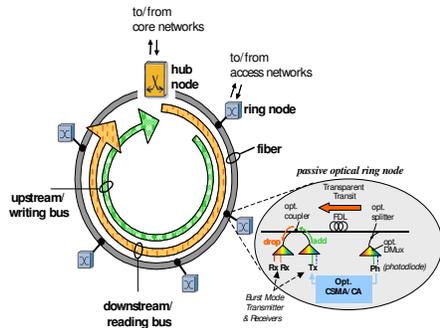


Fig. 1. Overview of DBORN network and node architecture

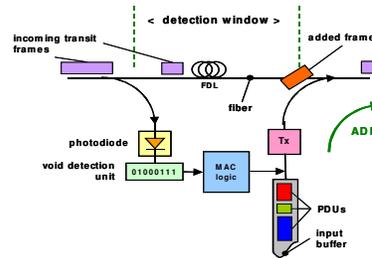


Fig. 2. Schema of the CSMA/CA based MAC of DBORN

2.1 DBORN Architecture

DBORN can be described as a unidirectional fiber split into downstream and upstream channels spectrally disjoint (i.e. on different wavelengths). The downstream bus, initiated at the hub node, is a medium shared for reading purposes, while the upstream bus, initiated in the ring nodes, is a multiple access-writing medium. Note that all the transmissions in the ring are performed between the hub and the ring nodes. Therefore, in the downstream direction, DBORN is a point-to-multipoint network, and in the upstream direction it is a multipoint-to-point network.

Let us consider N nodes placed in the unidirectional ring, as shown in Fig. 1. Each node serves one or more access networks. With regard to the direction from the access networks to the feeder ring, the ring node plays the role of a concentrator. Buffered packets are transmitted along the upstream bus toward the hub. In fact, packets travel along the ring without any electro-optic conversion at intermediate nodes. Thus we deal with the optical transparency property of the transit traffic through intermediate ring nodes. The hub terminates upstream wavelengths and electronically processes the packets. According to its destination, a packet is forwarded either into the backbone or through the downstream bus to reach the ring nodes which are being destined. In the latter case, the ring node picks up a copy of the downstream signal, originating from the hub, by virtue of a splitter in order to recover its corresponding packets. Once split, the main signal is no more processed at the node level and it continues its path to serve the other ring nodes, since a wavelength channel could be shared in reception by several nodes. However, the ring node terminating the copied channel, electronically processes the data packets contained therein, and delivers them to users.

2.2 MAC Design

In terms of logical performance, the main issue is related to the collision-free packet insertion on a shared writing bus. Since the path in transit remains transparent and

passive (neither an active optical device nor electronic conversion are employed to handle transit frames), no packet is dropped once transmitted on the ring. Hence, traffic control mechanisms are required at the electronic edge of the ring nodes to avoid collision with transit traffic during its own data emission.

In a fixed-slotted ring system with fixed packet size, void (i.e., slot) filling can be carried out immediately upon detection, since the void duration is a multiple of the fixed packet size duration. The detected void is therefore guaranteed to provide a minimum duration of one fixed packet length. However, in nonslotted ring systems with variable packet length and arbitrary void duration, it is very likely that a collision will occur if a packet is immediately transmitted upon detection of a void's edge.

To meet these requirements, each ring node must retain the upstream traffic flow within the optical layer while monitoring the medium activity. So, as shown in Fig. 2, each ring node first uses an optical splitter to separate the incoming signal into two identical parts: the main transit signal and its copy used for control purposes. With regard to the control part, as in [5], low bit rate photodiodes (ph) –typically 155 MHz– are used to monitor the activity of the upstream wavelengths. Once a free state of the medium is detected, the MAC unit measures the size of the progressing void.

To do so, a Fiber Delay Line (FDL) is introduced on the transit path to delay the upstream flow by one maximum frame duration augmented by the MAC processing time. The length of the FDL is slightly larger than the Maximum Transmission Unit (MTU) size allowed within the network, in order to provide the MAC unit with sufficient time to listen and to measure the medium occupancy. The ring node will begin injecting a packet to fill the void only if the null period is large enough (i.e. at least equal to the size of the packet to be inserted). Undelivered data will remain buffered in the electronic memory of the ring node until a sufficient void space is detected. This way, collision free packet insertion on the upstream bus from the add port is ensured.

However, considering only this basic mechanism, Head Of the Line (HOL) blocking and fairness issues arise. A direct result would be performance degradation for ring nodes that are close to the hub node on the upstream bus. Additional flow control mechanisms must be thus considered, both at the MAC layer and at the upper layers of edge nodes.

3 The Proposed Protocol: TCARD

To regulate the transmission of a node, we can use the TB rate-based control scheme put into action in the case of slotted WDM ring. However, as demonstrated in [4], this same mechanism presents several limitations when dealing with the fairness problem in the case of asynchronous transmission.

We showed in [4] that even if the traffic sourced by upstream nodes, which is regulated by the TB shaper, does not violate the negotiated throughput, it causes unacceptable packet delay to downstream nodes sharing the same channel. Certainly, thanks to TB algorithm the free bandwidth (stated in bit/s) allocated to each node is theoretically sufficient to handle its traffic. However the main issue pertains to the inappropriate distribution of the free bandwidth. This is mainly due to the lack of

organization during the emission process in the network. In fact, the mismatch, between the idle period distribution resulting from the upstream nodes' utilization of the medium and the packets' size distribution of the downstream node, often leads to bandwidth waste as well as fairness problems with regard to resource access.

To cope with the aforementioned factors, we suggest a new preventive control mechanism that forces a node to conform to its allocated bandwidth, and it also prevents the random division of the resource capacity. Let us consider N nodes sharing a common unidirectional channel traveling to the hub and let us consider more specifically the i^{th} node of the bus. Unlike the TB algorithm which is based on the i^{th} node traffic descriptor, the TCARD algorithm relies on the specification of the aggregate traffic sourced at the downstream nodes ($i+1, \dots, N$). The TCARD mechanism is based on the distribution and the gathering of remote information (descriptors) relative to the traffic requirements contracted by downstream ring nodes sharing the same resource. For each ring node $i (i = 1, \dots, N)$, the aggregate descriptor reflects the traffic needs of its downstream neighbors. This information is then used to constrain the access of the node i to the medium.

The basic operation of such a scheme is simple. To describe it, let us consider an anti-token pool at the node i where anti-tokens are generated at fixed time intervals that correspond to the specified average rate of the aggregate downstream traffic. Unlike TB scheme where each token represents the permission to transmit one bit, each TCARD anti-token prevents the node i from transmission on a detected idle period for a fixed amount of time S . This idle period of size S is reserved for downstream nodes. Following this reservation, an anti-token is removed from the anti-token pool. Packets arriving to the node enter first to the input buffer. Thanks to the MAC protocol, the node listens and measures the void duration present in the medium. If the detected null period is smaller than S , i.e. the anti-token cannot be placed, the void could be used by the node for its own usage purpose as long as the cumulative size of frames to be sent is smaller than the current void size. If the detected null period is equal or greater than S , it is used mainly for the release of one anti-token if the anti-token pool is not empty, otherwise the void can be used by the node for transmission. The arrival of an anti-token to the pool during the transmission of a packet does not preempt the emission process. The anti-token has to wait the packet service completion in order to be served.

Hence, the main idea behind TCARD lies in the preservation of bandwidth (represented by voids) by upstream nodes in order to satisfy downstream nodes' requirements. However, organized and sensible reservation schemes must be applied to fully benefit from the protocol efficiency. A basic rule consists in avoiding random division of the resource leading to inadequacy between the idle period length and the packet size distributions. Indeed, in order to guarantee for a downstream node the ability to transmit packets of maximum size, the length S of a reserved void (which consumes one anti-token) must be at least equal to the MTU size. On the other hand, the maximum length of an idle period that can be reserved is limited by the FDL length adopted in the MAC design. As stipulated earlier, the FDL length used is slightly larger than the MTU size allowed on the network. Therefore, to fulfill the

aforementioned requirements the duration S that has to be adopted must be equal to the time required to transmit the MTU on the shared medium.

4 Approximate Analytical Models of TCARD

In this section, we propose a first analysis of the TCARD mechanism. Our aim is to assess the mean waiting time $E[W_i]$ at each node $i(i = 1, \dots, N)$ when TCARD is enabled. To achieve this aim, we will primarily consider the first node. Furthermore, the analysis is extended to the following ring nodes.

4.1 Analysis of the First Node

The first node can be modeled by an M+D/G/1 queue with non-preemptive priority. In fact, the workload for the first queue consists of two classes of job: the anti-tokens, which represent the higher priority class, and the data flows (i.e. packets). Anti-tokens arrive to each node according to a deterministic process with an arrival rate τ_i and require a fixed service time S . On the other side, packets arrive according to a Poisson process with rate λ and require a general service time. Recall that the objective is to determine the average waiting time of the lower priority class jobs. This analysis is quite complicated so we propose an accurate approximation of the waiting time of the first node.

In this study, the queue server can be in one of two states: free or busy. The server is busy whenever an anti-token or a packet is present in the system, otherwise the server is free. Let us consider a packet $\#j$ entering the queue. Two cases are to be distinguished:

- a. When the packet arrives, it is the only job of the lower priority class. The packet transmission begins as soon as there are no higher priority jobs (anti-tokens) remaining in the queue. Hence the service time of the packet can be emulated as follows:
 1. If the queue server has gone in a free state since the latest packet transmission (i.e. packet $\#j-1$), the emulated service time includes the residual service time of the eventual anti-token already present in the server when the packet $\#j$ arrives. So packet $\#j$ may take a waiting time (set up time) before its transmission can start. Note that at most one anti-token may be present in this case, since the anti-token generation rate is lower than the medium throughput. The emulated service time gets over as soon as there are no anti-tokens remaining in the queue since the beginning of packet $\#j$ transmission.
 2. The queue server is always in a busy state since the packet $\#j-1$ has been successfully transmitted. In other words, the emulated service of packet $\#j-1$ is not yet completed. Once this service is completed (i.e. there is no more anti-tokens in the queue) the packet $\#j$ is taken immediately into service. In this case there is no waiting time or set up time. Indeed, the emulated

service of packet # j begins once the packet # j transmission starts and gets over as soon as there are no anti-tokens in the queue.

- b. In the second case, the packet arrives while other packets are still present in the system. The packet # j has to wait its turn to be served. When the service of packet # $j-1$ is completed, the packet # j is taken into service immediately. In this case, the emulated service time retains the same expression as in the case a-2 and there is not a set up time. Henceforth both cases will be aggregated in the remainder of this study.

In summary, this model behaves as an M/G/1 queue with a set up time [6]. For this model, we first derive the emulated service time distribution as described above. Moreover, we obtain the mean waiting time.

4.1.1 Emulated Service Time With Possible Non Null Set Up Time

It corresponds to the case a-1. As explained earlier, the packet # j has to wait during the residual service time of the eventual anti-token still present in the system. Then packet # j is transmitted. In this case, the emulated service of packet # $j-1$ is already completed. Thus the system is “empty” of customers when packet # j arrives even if the server may be occupied by the last generated anti-token. Let T_0 denote the arrival time of the packet # j . Due to the memoryless property of Poisson arrivals, we consider that the arrival time A_0 of the last generated anti-token is uniformly

distributed over $\left[T_0 - \frac{1}{\tau_1}, T_0 \right]$ where τ_1 is the generation rate of anti-tokens. Two

cases can be distinguished:

- Case 1: $A_0 < T_0 - S$, i.e. the last generated anti-token has already left the system when the packet # j arrives. The set up time is thus null. The packet # j transmission begins immediately (at $X_0 = T_0$) and will finish at time $X_0 + S_j$, where S_j is the transmission time of packet # j . The following anti-token will arrive at time $A_1 = A_0 + \frac{1}{\tau_1}$.
 - If $X_0 + S_j < A_1$, then the node can transmit a new packet at time $X_0 + S_j$. Hence the service time of the packet # j is simply S_j and the set-up time is equal to 0.
 - If $X_0 + S_j \geq A_1$, the anti-token will be sent just after the departure of the packet # j , during the time period $[X_0 + S_j, X_0 + S_j + S]$. The next anti-token will arrive at time $A_2 = A_0 + \frac{2}{\tau_1}$. The analysis is iterated until $X_0 + S_j + kS < A_{k+1}$ where $A_{k+1} = A_0 + \frac{k+1}{\tau_1}$. In this case, the set-up time is zero and the emulated service time equals to $S_j + kS$.

- Case 2: $A_0 > T_0 - S$, i.e. the last arrived anti-token is still in the system when the packet # j arrives.
The packet # j has to wait at least during the time period $[T_0, A_0 + S]$ before its transmission starts. This will constitute the set-up time of the system. As $S < \frac{1}{\tau_1}$, the last arrived anti-token will leave the queue before the arrival of the following one. Indeed, $A_0 + S < A_1 = A_0 + \frac{1}{\tau_1}$. Thus, the packet # j will enter the server at time $X_0 = A_0 + S$ and will be transmitted during the time period $[X_0, X_0 + S_j]$. Then, one falls down in the same study as in the first case with $X_0 = A_0 + S$ instead of $X_0 = T_0$. Hence the service time is given by the expression:

$$S_j + kS + X_0 - T_0 \text{ where } \begin{cases} X_0 + S_j + kS < A_{k+1}, \\ X_0 + S_j + (k-1)S > A_k \\ \text{and with } A_k = A_0 + \frac{k}{\tau_1} \end{cases} \quad (1)$$

The set up time is $X_0 - T_0$ in this case.

4.1.2 Emulated Service Time With Null Set Up Time

We refer here to the cases a-2 and b. As stated before, the service time expression remains the same in both cases. When the packet # j arrives, the emulated service of packet # $j-1$ is not yet completed. Thus the system is not “empty” of customers when packet # j arrives. The set up time is zero. The emulated service time starts at X_0 as soon as there is no anti-token in the system since the arrival of packet # j . Hence, the last anti-token arrived at A_0 with $X_0 - \frac{1}{\tau_1} < A_0 < X_0 - S$. We assume that the arrival

of the last anti-token is uniformly distributed between $\left[X_0 - \frac{1}{\tau_1}, X_0 - S \right]$. The packet # j will be emitted during the time period $[X_0, X_0 + S_j]$. If during this transmission period, the following anti-token arrives ($A_1 = A_0 + \frac{1}{\tau_1} < X_0 + S_j$), it will be served at time $X_0 + S_j$ and will leave the server at time $X_0 + S + S_j$. In a similar way as previously, this operation is iterated in order to derive the “emulated service time” of a packet # j .

4.1.3 The Mean Waiting Time for the First Node

The first node is then analyzed as an M/G/1 queue with set-up time [6]. Let C_0 (resp. C_1) denote the emulated service time when the system is “empty” (i.e. case a-1) (resp. not empty, i.e. cases a-2 and b) at arrival epochs. Let π_0 , be the probability that a packet arrives while the system is “empty” of customers. It can be shown that:

$$\pi_0 = \frac{1 - \lambda E[C_1]}{1 + \lambda E[C_0] - \lambda E[C_1]} \quad (2)$$

The mean “emulated service time” $E[C]$ (including set-up times) is consequently equal to:

$$E[C] = \pi_0 E[C_0] + (1 - \pi_0) E[C_1] \quad (3)$$

The mean response time can be expressed as follows:

$$E[R] = \pi_0 \left(\frac{2E[C_0] + \lambda E[C_0^2]}{2(1 - \lambda E[C_1])} + \frac{\lambda^2 E[C_0] E[C_1^2]}{2(1 - \lambda E[C_1])^2} \right) \quad (4)$$

The mean waiting time is thus equal to:

$$E[W_1] = E[R] - E[C] + \pi_0 \frac{S^2}{2} \tau_1 \quad (5)$$

Where $\frac{S^2}{2} \tau_1$ is the average residual service time of the eventual anti-token present in the server when a packet arrives while the system is “empty”.

Different packet length distributions can be considered. In the present work, we consider packets of variable length (50, 500 and 1500 bytes) more or less representative of the peaks in packet size distribution in Ethernet. The total traffic volume comprises 50% of 1500 Bytes, 40% of 500 Bytes and 10% of 50 Bytes packets size. Let p_i be the probability of the different packet sizes and d_i the corresponding emission time. The transmission time of an anti-token S is chosen equal to the emission time of the longest packets.

For each type i of packets, we get the probability density function of $C_{1,i}$ (index i refers to packets of length d_i):

$$f_{C_{1,i}}(x) = \frac{d_i - k \left(\frac{1}{\tau_1} - S \right)}{\frac{1}{\tau_1} - S} \delta(x - (d_i + (k+1)S)) + \frac{\frac{1}{\tau_1} - S - d_i - k \left(\frac{1}{\tau_1} - S \right)}{\frac{1}{\tau_1} - S} \delta(x - (d_i + kS)) \quad (6)$$

where δ denote the Delta Dirac function and $k = \left\lfloor \frac{d_i}{1/\tau_1 - S} \right\rfloor$.

Which yields to:

$$\left\{ \begin{array}{l} E[C_{1,i}] = \frac{d_i - k\left(\frac{1}{\tau_1} - S\right)}{\frac{1}{\tau_1} - S} (d_i + (k+1)S) + \frac{\frac{1}{\tau_1} - S - d_i - k\left(\frac{1}{\tau_1} - S\right)}{\frac{1}{\tau_1} - S} (d_i + kS) \\ E[C_{1,i}^2] = \frac{d_i - k\left(\frac{1}{\tau_1} - S\right)}{\frac{1}{\tau_1} - S} (d_i + (k+1)S)^2 + \frac{\frac{1}{\tau_1} - S - d_i - k\left(\frac{1}{\tau_1} - S\right)}{\frac{1}{\tau_1} - S} (d_i + kS)^2 \end{array} \right. \quad (7)$$

In a similar way, the probability density function of $C_{0,i}$ is given by:

$$f_{C_{0,i}}(x) = f_{C_{1,i}}(x) \frac{\frac{1}{\tau_1} - S}{\frac{1}{\tau_1}} + \frac{(x + d_i + kS)}{\frac{1}{\tau_1}} 1_{\{-S < x < 0\}} \quad (8)$$

The first two moments of those service times are then equal to:

$$\left\{ \begin{array}{l} E[C_0] = \sum_i p_i E[C_{0,i}] \quad E[C_0^2] = \sum_i p_i E[C_{0,i}^2] \\ E[C_1] = \sum_i p_i E[C_{1,i}] \quad E[C_1^2] = \sum_i p_i E[C_{1,i}^2] \end{array} \right. \quad (9)$$

4.2 Analysis of the Following Nodes

The analytical model described so far handles the first node case. However, it can be easily extended to the following ring nodes without major modifications but still approximations have to be accounted for. To achieve this aim, we assume that the available bandwidth received by each node i is equal to the bandwidth reserved by upstream nodes due to the anti-tokens. Thus the server speed D_i of each node i is considered equal to:

$$D_i = S\tau_{i-1} \text{ with } i > 1 \quad (10)$$

Afterwards, we can recursively apply exactly the same method presented in section 4.1.

5. Performance Evaluation

In this section, we evaluate the performance of the proposed fairness control protocol. We consider a ring of 8 nodes sharing the same wavelength that runs at 1 Gbit/s. Each

node receives traffic from the access networks to be forwarded toward the hub at a mean rate of 0,1 Gbit/s. Thus the traffic sourced by all the ring nodes represents 80% of the wavelength capacity. The packets arrive according to a Poisson process. The arrival rate of the packets at each node is the same in order to highlight the fairness issues. In this section we assess the access delay and the packet loss rate (PLR) at each ring node. As explained before, the anti-token generation at each node within TCARD is configured to reflect the average amount of traffic expected at downstream nodes. Moreover, the bandwidth reserved for downstream nodes' use purpose is representative of voids of 1500 bytes in order to comply with packets of maximum size. For instance, the TCARD anti-tokens are generated at the first node at a rate of $\tau_1 = (0,1 \cdot 7 \cdot 10^9) / (1500 \cdot 8)$ anti-tokens/s.

We first compare the performance of each ring node whether TCARD is enabled or not. Then, we evaluate the accuracy of the proposed analytical model by comparing its results with those obtained from simulations carried out using network simulator 2. Figure 3 depicts the average access delay experienced by packets arriving to each node. Results highlight the fairness issue already discussed. We point out that the performance degradation, when TCARD is disabled, is not due to the medium saturation since the channel occupancy is below 70 %. This is mainly due to the lack of organization during the emission process in the network. In fact, the mismatch, between the idle period distribution resulting from the upstream nodes' utilization of the medium and the packets' size distribution of the downstream node, often leads to bandwidth waste as well as fairness problems with regard to resource access. Recall that the input load is 80%. This difference is simply due to the packet loss resulting from buffer overflow of downstream nodes.

On the other hand, TCARD enables fairness and better use of the resource by sharing efficiently the bandwidth between competing nodes. The mean access delay is around 160 μ s for all the nodes. We notice that the performance of the downstream nodes when going closer to the hub is not affected by the upstream nodes. In addition, simulations show that TCARD improves the resource utilization, which increases from 70% to 80%. We point out however that the delay recorded at upstream nodes is slightly increased, with respect to the case where no fairness control mechanism is

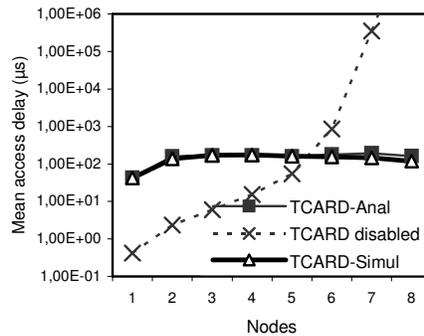


Fig. 3. Mean access delay of the eight-node bus with variable-packet size traffic

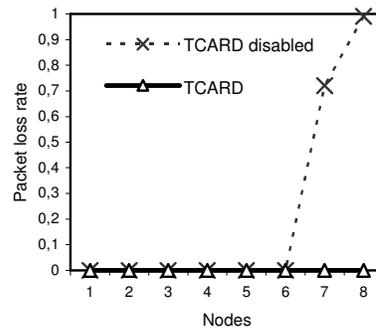


Fig. 4. Packet loss rate of the eight-node bus with variable-packet size traffic

applied, but still remains below 170 μ s. This is because TCARD algorithm imposes more constraints on upstream nodes in order to preserve usable bandwidth for downstream ones. The analytical results of access delay when TCARD is enabled reveal a perfect match with the simulation results: analytical results practically coincide with the simulation results. So, the approximate analytical models can achieve high accuracy.

Figure 4 depicts the PLR at each ring node. The capacity of the electronic buffer at each node is set to 1 Mbytes. As expected, when TCARD is disabled, packet loss occurs more and more when approaching the hub due to the node buffer overflow. Indeed, downstream nodes do not find suitable idle period to transmit their packets. In particular, the loss rate registered at node 8 is above 99% in the absence of any control mechanism. In contrast, with TCARD, no packet loss is recorded in the network. In this case TCARD never incurs loss due to its efficient share of bandwidth among nodes.

6 Conclusion

This paper provides the analysis of shared bus network's behavior with asynchronous transmission. We analyzed the system performance in terms of access delay required by each node to inject a packet on the shared medium. The analysis results showed that fairness issues are likely to arise between upstream and downstream nodes. To alleviate the fairness problem, we suggested a new protocol, called TCARD, based on a preventive reservation scheme. We developed analytical models in order to assess the access delay at each node when TCARD is enabled. The proposed models are proven to be highly accurate by comparison with simulation results. The analytical and simulation results showed how the proposed solution alleviates the performance degradation and the resource sub-utilization while achieving fairness among bus nodes.

References

1. N. Le Sauze et al.: A novel, low cost optical packet metropolitan ring architecture. Proc. of ECOC '01, Amsterdam, Netherlands, Vol. 4 (October 2001) 66-67
2. M. A. Marsan, A. Bianco, E. Leonardi, F. Neri, and S. Toniolo: Metering Fairness Control Schemes in All-Optical WDM Rings. Proc. of INFOCOM '97, Kobe, Japan, vol. 2 (April 1997) 752-760
3. N. Bouabdallah, A. L. Beylot, G. Pujolle: Fairness Issues in Bus-Based Optical Access Networks. Proc. of Networking '04, Athens, (May 2004)
4. N. Bouabdallah et al.: Resolving the Fairness Issue in Bus-Based Optical Access Networks. IEEE Commun. Mag., vol. 42, no. 11, (Nov. 2004) pp. 2-8
5. R. Gaudino et al.: RINGO: a WDM Ring Optical Packet Network Demonstrator. Proc. of ECOC '01, Amsterdam, Netherlands, Vol. 4 (September 2001) 620-621
6. H. Takagi: Queueing Analysis Vol I: Vacation and Priority Systems Part I. North Holland (1991)