



ELSEVIER

Journal of Systems Architecture 46 (2000) 1115–1135

JOURNAL OF
SYSTEMS
ARCHITECTURE

www.elsevier.com/locate/sysarc

Design of a large-scale Gbit/s MAN using a cyclic reservation-based MAC protocol

Wen-Fong Wang^{a,*}, Wen-Shyang Hwang^b, Jun-Yao Wang^c

^a Kaohsiung P.O. Box 742, Kaohsiung city 800, Taiwan, ROC

^b Department of Electrical Engineering, National Kaohsiung Institute of Technology, Kaohsiung 807, Taiwan, ROC

^c Computer Laboratory, Department of Electrical Engineering, National Cheng-Kung University, Tainan 701, Taiwan, ROC

Received 13 February 1999; received in revised form 27 May 1999; accepted 8 April 2000

Abstract

In this paper, a large-scale Gbit/s metropolitan area network (MAN) based on hierarchical ring topologies has been investigated. The network is constituted by backbone and local rings, which are connected by bridges. In the network, traffic congestion may occur in bridges due to the mismatch of transmission speed between backbone and local rings. To cope with the issue, a scheme based on a cyclic reservation-based access method is proposed for the medium access control (MAC) protocol of the network. With the scheme, the protocol can both achieve the fair access of network bandwidth and resolve the traffic congestion in bridges. Particularly, this approach reduces the implementation complexity and cost and enhances efficiency of the global network. Finally, several simulative experiments are performed, and some optimistic results are revealed. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: High-speed LANs/MANs; Gbit/s networks; Hierarchical ring topology; Cyclic reservation-based MAC protocol; Fairness scheduling

1. Introduction

Other than local or wide area networks (LANs/WANs), metropolitan area networks (MANs) are expected simultaneously to connect a great number of dispersed users together and support them with high-speed communications. Today, well-known standards to build MANs are the *fibre distributed data interface* (FDDI) and the *distributed queue dual bus* (DQDB) [17,18]. They are built with modest methods, existing technology, and linear topologies and operate at the speed of 100 Dbits/s and a few hundreds in the case of DQDB. Recently, as the rapid development of new applications, such as multimedia, world-wide web, virtual reality, voice and video services on Internet, etc., more and more users within a campus or a big community are eager to use high-speed networks. Thus, those existing

* Corresponding author. Fax: +886-7-344-3861.

E-mail address: wwf@ms.chttl.com.tw (W.-F. Wang).

networks are inefficient to cope with the user requirements. The trend is the development of MANs with larger size and higher bandwidth.

To design a MAN, three fundamental issues need to be considered: transmission media, network topologies, and medium access control (MAC) methods. With the advent of optical fiber as transmission media, Gbit or even Tbit communications are feasible. In this study, assume optical fiber as basic transmission media and Gbit/s as basic rate. For network topologies, several advantages can be found for the networks with higher connectivity [1]. First, large-size networks can be built to connect more users than traditional MANs like FDDI and DQDB. Next, extremely high aggregate network bandwidth can be achieved due to more links used in network spans. A remarkable network presented in the literature is a grid network, e.g., the *Manhattan Street Network (MS_Net)* [11,12]. It has high connectivity, but suffers from a variety of problems [5,13]. Obviously, it is difficult and costly to realize this kind of networks for commercial applications. Considering another case, a network with the hierarchical ring topology shown in Fig. 1 is feasible. The network is called the *hierarchical ring* and consists of a number of local rings, which are connected by the backbone ring with bridges. Based on slotted and buffer insertion access mechanisms, it has the advantages of simple routing and straightforward media access. Particularly, its topology has the property of self-similarity, i.e., ring connecting ring, and thus can extend easily to a multiple-level hierarchy [9]. In the hierarchical rings, a bridge is used to switch the traffic flows to/from some other local rings (called *inter-traffic*) or within the same local ring (called *intra-traffic*). However, such networks may have traffic congestion in a bridge [7]. This situation occurs owing to the difference of network bandwidth between the higher-speed backbone ring and lower-speed local rings, and owing to unbalanced workloads from different inter-traffic flows. For instance, as one inter-traffic flow becomes heavy, a long queue of packets is built up in the destination bridge and some packets are eventually lost due to buffer overflow. Considerable congestion also arises when packets come from different bridges gather on a certain bridge simultaneously. To resolve the congestion, a buffer control mechanism for inter-traffic flows must be included in the MAC protocol for the hierarchical rings.

As for access control methods of the network, we propose a protocol based on a cyclic reservation-based MAC scheme, named as the *cyclic reservation multiple access for hierarchical rings (CRMA-HR)*. Originally, the cyclic reservation-based MAC scheme was proposed by IBM and had two versions

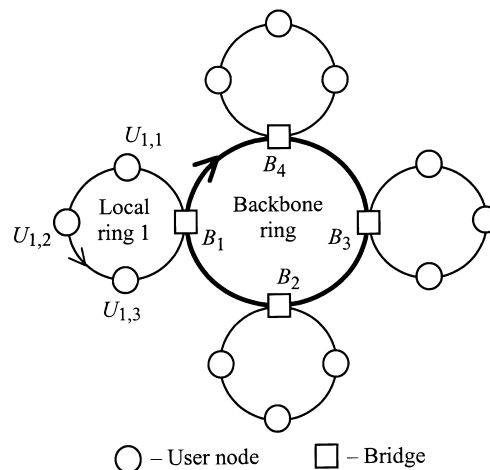


Fig. 1. The topology of a 4×4 single hierarchical ring.

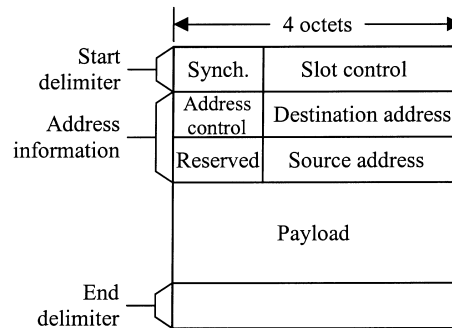
developed, i.e., cyclic reservation multiple access (CRMA) [15] and CRMA-II (version 2 of CRMA) [3,4,10]. The latter version, namely, CRMA-II, was the enhancement of CRMA for bus and ring topologies and had been shown to be rather efficient in comparison with other MAC protocols for high-speed LANs and MANs [1,8,14,16]. Remarkably, these protocols satisfy fairness requirements [2,6] so that the properties of high network throughput and low response time are sustained under heavy network loads. Comparing with the above two protocols, CRMA-HR is more complex. First of all, it is used for the hierarchical rings with bridges containing network buffers. Second, the avoidance of traffic congestion and the management of network buffers must be considered together. From communication viewpoints, buffer capacity is as important as network bandwidth in the hierarchical rings. They must be treated as different kinds of network resources since both of them are finite for users using at a given time and mandatory for communications. Thirdly, CRMA-HR has to control the resources inclusively rather than the network bandwidth solely. Accordingly, CRMA-HR is composed of two parts: the *bandwidth control protocol* (BwCP) and the *buffer control protocol* (BfCP). For BwCP, it behaves similar to CRMA-II. Initially, a network node executing BwCP can identify itself as the role of an initiator or a responder while the network is heavily loaded. If it is an initiator, it starts a series of cyclic reservation cycles for the fair access of network bandwidth. At this point, those remaining nodes (being responders) would cope with the procedure in reservation cycles to share the bandwidth fairly. For BfCP, it has similar steps as BwCP to control traffic through bridges and prevent possible network congestion due to buffer overflow. Using a *fairness threshold computation algorithm*, it can determine a threshold value for the fair usage of buffer capacity among different inter-traffic flows while the buffer is heavily loaded. Note that BwCP and BfCP use similar mechanisms for the access control of different resources. As a result, the implementation complexity of CRMA-HR can be reduced significantly.

For the network, CRMA-HR satisfies the requirements of high throughput, fairness, implementation simplicity, and scalability in terms of geographical coverage and number of attachments. Particularly, the problem of network congestion is resolved using a fairness buffer control mechanism. In Section 2, a network model associated with some basic access mechanisms for the network resources is briefly sketched for the hierarchical rings. In Section 3, the two parts of CRMA-HR for network bandwidth and buffers are described. Due to the resources existing in the network, the reservation cycles for different resources may occur simultaneously. In addition, due to the traffic flows interleaving from one bridge, a head-of-line blocking phenomenon may occur. These issues will be discussed. To evaluate the preliminary performance properties, simulation models and network configurations for CRMA-HR are addressed in Section 4. Subsequently, several simulative experiments under different traffic conditions are done in Section 5. Finally, a few remarks are given in Section 6.

2. Network architecture

The single hierarchical ring shown in Fig. 1 consists of a number of local rings, which can connect a large number of *user nodes*. The local rings are then connected to the higher bandwidth backbone ring via *bridges*. Notably, each subnetwork, namely, the backbone ring or one of the local rings, can be replaced by a dual ring. Thus, some variations may exist, e.g., a dual backbone ring connecting single local rings, all subnetworks being dual rings, and so forth. For simplicity, we consider the single hierarchical ring throughout this paper.

For CRMA-HR, a slotted transmission structure is adopted with the slot being the unit of reservation. In Fig. 2, the slot format is shown. It consists of the start and end delimiter pair with the address information and payload embedded [3]. The start delimiter and the address information form the header of a slot. For the start delimiter, it consists of the synchronization part and the slot control part, which contains



Slot control:

Slot_type – single slot/multi-slots/command slot
 Reservation_cycle – none/bandwidth/buffer
 Status – Busy/Free flag, Gratis/Reserved flag
 Concatenation – First/Middle/Last or Only
 Priority, Monitor

Address control:

Hierarchical_level, Address_type

Fig. 2. The slot format.

Slot_type, Reservation_cycle, Status, Priority, Monitor, and Concatenation. The Slot_type field is used to identify a transmission entity as a slot, as a multi-slot, or as a command slot. The Reservation_cycle field is used to indicate none, the cycles for bandwidth, or the cycles for buffers. The Status field of a slot used in media access control is given by two flags, the *Busy/Free* flag and the *Gratis/Reserved* flag. Users have unrestricted access right to the *Free-Gratis* slots and then mark them as the *Busy-Gratis* slots while using. If a slot is reserved in a reservation cycle, it is marked as a *Free-Reserved* slot. The Concatenation field is used to specify whether a slot is single (e.g., an ATM cell) or is part of a multi-slot frame; in latter case its position is further defined as first, middle or last. On a subnetwork, the Monitor field is used for detecting and freeing circulated slots.

The address scheme for the hierarchical rings is more complex than the schemes for single or dual ring networks. It consists of the parts of the address control, destination address, and source address (see Fig. 2) and can provide larger address space. Although the addresses with total 8 octets in each slot consume a little extra network bandwidth, this scheme can support addressing a large number of network nodes. The address control part contains the fields of Hierarchical_level and Address_type. The Hierarchical_level field is used to indicate the level number of the network hierarchy for the hierarchical rings. According to this level number, the Address_type field is used to extract the address of a bridge or a user node. For instance, given Hierarchical_level 2 bits and Address_type 6 bits, respectively, the destination/source address with the field length of 3 octets (or equivalently 24 bits) can be divided into six 4-bits groups. If Hierarchical_level is 2 and Address_type is 000111, then the upper three groups are for the address of bridges in the backbone ring and the lower three groups are for the address of user nodes in local rings. By the hexadecimal format, the address of a bridge can be denoted as $X_1X_2X_3000$ (e.g., 1A7000), and the address of a user node in the corresponding local ring as $X_1X_2X_3X_4X_5X_6$ (e.g., 1A7025). Similarly, suppose that Hierarchical_level is 3 and Address_type is 110011. Then, the first two groups are for the address of bridges in the level 2 backbone ring, the second two groups for the address of bridges in level 1 backbone rings, and the last two groups for the address of user nodes in local rings. By this way, the address scheme can be applied to various address types and multiple-level hierarchical rings.

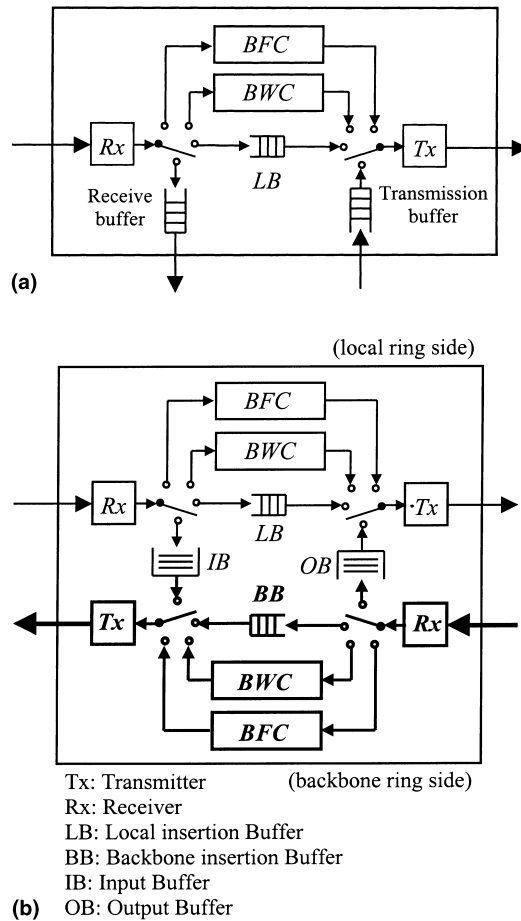


Fig. 3. The structures of (a) the user node and (b) the bridge.

A user node (Fig. 3(a)) is used to connect user equipment. Based on slotted and buffer insertion access mechanisms [3], it consists of a receiver, transmitter, receive buffer, transmission buffer, and insertion buffer. It also includes a bandwidth controller (*BWC*) and a buffer controller (*BFC*), executing BwCP and BfCP, respectively. *BWC* operates at two states, namely, passive and active. At the passive state, it employs a *simple busy/free mechanism* [3,4] for direct accessing the transmission medium when the network is lightly loaded. Media access for packets in the transmission buffer is only allowed, if there is a *Free-Gratis* slot arriving to a user node and its insertion buffer is empty. Slots may be used several times by spatial reuse because of the destination release strategy. Transmission of a multi-slot packet is supported by delaying incoming slots in the insertion buffer to allow the transmission of a complete packet in each node. When user nodes suffer from poor access due to heavy network loads, *BWC* switches to the active state. At this point, the cyclic reservation scheme is employed to yield a sustained high network utilization while maintaining fairness and tight access delay bounds. At a given time, only one *BWC* can be in the active state. For *BFC*, we describe it subsequently.

A bridge (Fig. 3(b)) is used to connect a local ring to the backbone ring. It consists of two sets consisting of a transmitter, receiver, insertion buffer, and *BWC*s for the local and backbone rings,

respectively. Additionally, it includes two *BFCs* and two buffers for incoming and outgoing inter-traffic. Due to the identical media access control mechanisms in local and backbone rings, the two *BWCs* in a bridge are the same, except the *BWC* for the backbone ring having higher processing speed. For the inter-traffic buffers, one stores packets from a local ring to the backbone ring and the other stores packets in the reverse direction. The former and the later are called the *input buffer (IB)* and the *output buffer (OB)*, respectively. To prevent buffer overflow and to achieve the fair usage of buffer capacity, *BFC* is used to control the utilization of *IB* or *OB*. As *BWC*, *BFC* has two states. If the input rate to a buffer (*IB* or *OB*) is less than or equal to its output service rate and the backlog of slots at that buffer is not built up, the capacity utilization of the buffer will be stable. At this point, its corresponding *BFC* is doing nothing but continuously monitoring the remaining buffer capacity in the passive state. Otherwise, *BFC* switches to the active state and regulates the input traffic fairly by starting a series of cyclic reservation cycles.

Based on the above bridge structure, constructing a multiple-level hierarchical ring is feasible. For example, suppose there are n 2-level single hierarchical rings. A bridge can be used to connect a 2-level single hierarchical ring to the higher-level backbone ring so that a 3-level single hierarchical ring is constructed. Accordingly, the construction of multiple level hierarchy associated with dual ring variations is feasible as well.

3. Reservation-based control protocols

In this section, the method of cyclic reservation-based access control to network resources is described in advance. Subsequently, BwCP and BICP, the two parts of CRMA-HR, are discussed. Since these two control protocols operate independently, reservation cycles for different resources may occur simultaneously. Thus, how to coordinate these cycles is discussed. Additionally, a head-of-line blocking phenomenon may happen in case of several inter-traffic flows interleaving from a source bridge to the destination bridges that are heavily loaded. How to eliminate the phenomenon is addressed as well.

3.1. Generic access control to the resources

As described above, network bandwidth and buffer capacity are two different resources, which are necessary to the communications in the hierarchical rings. To arbitrate the utilization of resources under heavy network loads, the resources allocated to network nodes are done by means of cyclic reservation. The method is organized into cycles of three phases: *reservation*, *scheduling*, and *confirmation*. To aid the operations in each working phase, the necessary information includes command slots, data slots, counters, and variables. Command slots inside of the reservation and confirmation phases are the reserve and confirm slots, abbreviated to RSV and CFM, respectively. A RSV is used to collect the information for resource request and reservation from all nodes within a subnetwork in the reservation phase. A CFM, sent at once after the scheduling phase, carries the scheduling result so that all nodes are informed about the decision of fair usage of the resources. Data slots include the *Free-Gratis* slot (FG), the *Busy-Gratis* slot (BG), and the *Free-Reserved* slot (FR). Depending on the using status, a data slot can be a FG, BG or FR slot. Being a FR slot, it is used for allocating the resources. Counters include the transmission and request counters, abbreviated to *TC* and *RC*, respectively. For *RC*, it is increased according to the number of new packets entering the transmission buffer. While for every packet being transmitted, *RC* is decreased and *TC* is increased accordingly. *TC* and *RC* are often treated as a pair. For different resources, different pairs of counters are employed. The variables used include fairness threshold (abbreviated to *FT*), *Reservation_cycle*, *Bw_short_flag*, *Bf_short_flag*, and *H*. The value of *FT*

is computed by the fairness threshold computation algorithms in the scheduling phase and then carried by CFMs for broadcasting. For Reservation_cycle, it is the same as the slot field Reservation_cycle in the incoming slots and has the value of either *none*, *bandwidth* or *buffer*. Note that, for a command slot, the value of this field is either *bandwidth* or *buffer*. Bw_short_flag and Bf_short_flag are two boolean variables for the reservation cycles of bandwidth and buffer, respectively. They are set to *true* while the corresponding cycle for bandwidth or buffer occurs and reset to *false* as the cycle ends. The variable *H* is used to count the total number of slots reserved. It is also computed in the fairness threshold computation algorithms.

In the following algorithms, the type of an arriving or leaving slot is denoted by the slot attached with a superscript ‘-’ or ‘+’, respectively. ‘Slot@data’ means that the data is copied into the slot. The symbol ‘→’ means ‘give’. To denote a field or a field value in one slot, the expression ‘field.Slot’ is employed. For instance, the value of *FT* in CFM is denoted as ‘*FT.CFM*’. Additionally, there are a few simple functions used such as CopyToMemory, Compute_bw_FT, Compute_bf_FT, Remove, and MarkReserve. Their meanings are rather intuitive from their naming and will not explain further. To ease the subsequent descriptions, assume that there are *n* nodes in a subnetwork (i.e., $(n-1)$ user nodes and one bridge for local rings or *n* bridges for the backbone ring) and every packet transmitted occupies one slot exactly. In normal situation, the resource usage by one node has no restriction. Given that a node, e.g., node *k*, detects the shortage of either bandwidth or buffer capacity, node *k* (being the initiator) executes the initiator algorithm to start the reservation phase. Simultaneously, a RSV is issued around the subnetwork (step I1). For the other nodes, they execute the responder algorithm as the responders while the RSV being received. At this point, the information for resource request counted by the pair (*TC*, *RC*) is copied into the payload of RSV (step R1).

Initiator Algorithm (initiator *k*).

- ```

(I1) if (Bw_short_flag=true) /*issue RSV for bandwidth shortage*/
 RSV ⊕ (Reservation_cycle=bandwidth) → RSV+;
 if (Bf_short_flag=true) /*issue RSV for buffer shortage*/
 RSV ⊕ (Reservation_cycle=buffer) → RSV+;
(I2) if (RSV-)
 CopyToMemory[((TCi, RCi)i=1..n, i≠k].RSV];
(I3) if (Reservation_cycle.RSV=bandwidth)
 Compute_bw_FT[] → (FT, H); /*call bandwidth FT comput algorithm*/
 if (Reservation_cycle.RSV=buffer)
 Compute_bf_FT[] → (FT, H); /*call buffer FT comput algorithm*/
 Remove[RSV];
(I4) CFM ⊕ FT → CFM+;
(I5) while (H > 0) do {
 H=H - 1;
 if ((FT - TCk) > 0) {
 FG- ⊕ packet → BG+; /*confirmations for initiator*/
 TCk = TCk + 1; RCk = RCk - 1; /*update counters*/
 } else MarkReserve[FG-] → FR+; /*resource allocation*/
 }
 TCk = 0; /*reset transmission counter*/
 }
(I6) if (CFM-) Remove[CFM];

```

**Responder algorithm (responder  $i$ ).**

```

(R1) if $((RSV^-) \wedge (RC_i > 0) \wedge (i \neq k))$ {
 Reservation_cycle = Reservation_cycle.RSV;
 $RSV \oplus (TC_i, RC_i) \rightarrow RSV^+$;
} else RSV^+ ;
if $((FG^-) \wedge (Reservation_cycle = \text{buffer}))$
 FG^+ ; /*avoid the buffer overflow and suspend the free slot's access*/
(R2) if $((CFM^-) \wedge (i \neq k))$ {
 $FT = FT.CFM$; CFM^+ ;
(R3) $L = |FT - TC_i|$;
 while $(L > 0)$ do {
 $L = L - 1$;
 if $((FT - TC_i) > 0)$ {
 $FR^- \oplus \text{packet} \rightarrow BG^+$; /*confirmations for responder*/
 $TC_i = TC_i + 1$; $RC_i = RC_i - 1$; /*update counters*/
 } else $FG^- \rightarrow FG^+$; /*deferments of resource usage for fairness*/
 }
 $TC_i = 0$; /*reset transmission counter*/
}

```

In the scheduling phase, the initiator does the scheduling operation. When the RSV returns, node  $k$  copies the pairs  $(TC_i, RC_i)_{i=1..n, i \neq k}$  to the memory and computes the  $FT$  value (steps I2 and I3). The  $FT$  value is computed by invoking either the bandwidth  $FT$  computation algorithm or the buffer  $FT$  computation algorithm. As the scheduling is in progress, the network can be fully utilized by allowing the nodes to access FG slots. Nevertheless, this cannot be allowed to the reservation cycle for buffers due to the prevention of buffer overflow. As soon as the scheduling terminates, the confirmation process starts. The initiator issues a CFM with the  $FT$  value (step I4). Then, following the CFM, it marks a number ( $H$ ) of slots as reserved by setting the Gratis/Reserved flag to Reserved (step I5). For the other nodes (i.e., those responders), the  $FT$  value is copied into their memory (step R2) while the CFM passing. Subsequently, each responder takes a fairness action to determine whether it has the *confirmations* to access FR slots or the *deferments* to throttle the access of succeeding FG slots (step R3). That is, if  $FT > TC_i$ , the confirmation is  $(FT - TC_i)$  FR slots used to transmit new packets. Otherwise, the deferment is set to  $(TC_i - FT)$  FG slots so that the access opportunity of those slots is left to the downstream nodes. Finally, the initiator removes the CFM from the network (step I6). Note that if the situation of resource shortage is continued, the initiator can start another cycle again.

### 3.2. Bandwidth control protocol

The BwCP protocol includes the functions according to the initiator, responder, and bandwidth  $FT$  computation algorithms for the access control of network bandwidth of a subnetwork. The reservation cycle here is called the *bandwidth cycle*. In a subnetwork, the *BWC* of every node always monitors the utilization of network bandwidth. While a node detecting the poor access to the network bandwidth, it switches to the active state immediately and executes the initiator algorithm with *Reservation\_cycle* set to *bandwidth*. When the initiator generates a RSV, the *BWC* of the other nodes receiving the RSV will switch to the active state as well and execute the responder algorithm. If more than one *BWC* declare itself as the initiator at a time, the arbitration must be taken. Although a number of arbitrating strategies might exist,



the simplest way of arbitration is to compare their MAC addresses. That is, if an initiator receives another different RSV with larger MAC address, then it must change itself as the responder at once and drop the issued RSV. By this way, there is only one initiator existing and the other nodes are the responders.

As to the computation of a  $FT$  value, the algorithm described in the following is based on the principle of computing the minimum summation of either confirmations or deferments over all nodes in a subnetwork. Since it causes a minimal degree of control intervention for the nodes, this aids the generation of maximum throughput in the next reservation cycle. With this principle, the optimal threshold (denoted by the variable  $Topt$ ) is computed for the  $FT$  value between the maximum and minimum of the set  $\{TC_1, TC_2, \dots, TC_n\}$  by iterations. When the  $FT$  value is determined, the total number of slots reserved, i.e., the value of  $H$ , is computed as well. The values of  $FT$  and  $H$  are then returned to the initiator algorithm.

### Bandwidth $FT$ computation algorithm.

```

 $Tmax = \max\{TC_1, TC_2, \dots, TC_n\};$
 $Tmin = \min\{TC_1, TC_2, \dots, TC_n\};$
 $sum = \text{infinity};$
for ($i = Tmin, i \leq Tmax, i = i + 1$) { /*compute the optimal threshold (FT)*/
 $temp = \sum_{j=1}^n |(i - TC_j)|;$ /*compute the sum of either confirmation or deferment*/
 if ($temp < sum$) {
 $sum = temp; Topt = i;$ /*find the optimal threshold*/
 }
}
 $FT = Topt; H = 0;$
for ($i = 1, i \leq n, i = i + 1$) { /*compute the total number (H) of slots reserved*/
 if ($Topt > TC_i$) {
 if ($(Topt - TC_i) > RC_i$)
 $H = H + RC_i;$
 else $H = H + (Topt - TC_i);$
 }
}
return(FT, H);

```

To illustrate the computation of  $FT$  and  $H$  using the algorithm and the occurrence of bandwidth cycles, we take the scenario given in Fig. 4 as an example. Assume that  $U_{1,1}$  in Local ring 1 detects the condition of insufficient bandwidth. As shown in Fig. 5(a), the  $BWC$  of  $U_{1,1}$  switches to the active state and issues a RSV

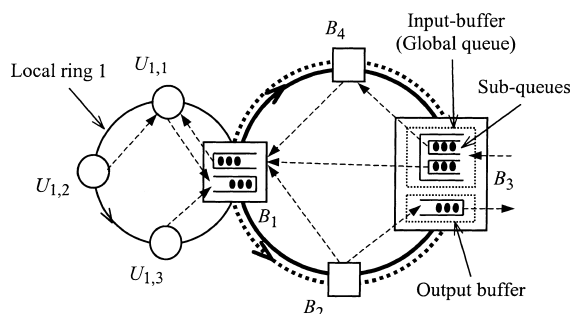


Fig. 4. An operational scenario for the bandwidth and the buffer reservation mechanism.

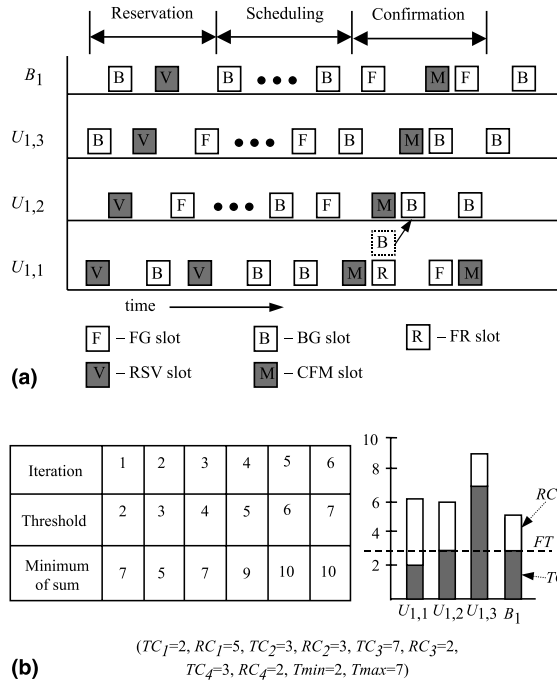


Fig. 5. (a) A bandwidth cycle; (b) a scheduling computation for bandwidth.

to collect the parameter pairs  $(TC_i, RC_i)$ , where  $i = 2, 3$  or  $4$ . For convenience, the pair  $(TC_4, RC_4)$  is used to represent the value pair of counters for  $B_1$ . In Fig. 5(b), the scheduling process and the values of parameters are given. After six computing iterations from  $T_{min} = 2$  to  $T_{max} = 7$ , the  $FT$  value is 3 and then the number of slots reserved ( $H$ ) is 1. At this point,  $U_{1,1}$  issues a CFM with  $FT = 3$  and marks one FG slot as a FR slot (Fig. 5(a)). However, the FR slot is used immediately by  $U_{1,1}$  itself and therefore changed to a BG slot.

### 3.3. Buffer control protocol

In bridges, buffer overflow due to heavy loads or overload conditions is critical if no proper congestion control is applied. In a hierarchical ring network, the BfCP protocol is used to control the inter-traffic flows through buffers ( $IB$  and  $OB$ ). As similar to BwCP, it provides the functions according to the initiator, responder, and buffer  $FT$  computation algorithms. Its reservation cycle is called the *buffer cycle*. Usually,  $BFCs$  for the  $IB$  and  $OB$  of every bridge always monitor the utilization of the buffer capacity counted in number of slots. Suppose that, in the backbone ring, the  $BFC$  for the  $OB$  of a bridge (say bridge  $B_k$ ) detects that the available buffer capacity reaches a level causing traffic congestion. It must then switch to the active mode (starting a buffer cycle by setting  $Reservation\_cycle$  to *buffer* and being the initiator) and issues a RSV command slot. When the  $BFC$  of bridge  $B_i$ , where  $i \neq k$ , receives the RSV, it executes the responder function and copies the pair  $(TC_i^k, RC_i^k)$  used to trace the inter-traffic from  $B_i$  to the initiator to the passing RSV. Subsequently, the RSV is reissued to downstream nodes. Note that, in the backbone ring, a few buffer cycles initiated by different  $BFCs$  can happen simultaneously without mutual interference. In particular, the buffer cycles will not exclude bandwidth cycles as well. Nevertheless, a special situation exists in local rings. Since a user node has no buffers like  $IB$  or  $OB$ , the only buffer related to the node is the  $IB$  in the corre-

sponding bridge (see the user nodes in Local ring 1 and  $B_1$  in Fig. 4). Therefore, the *BFC* of the user node is only necessary to execute the responder function (being a permanent responder) while receiving the *RSV* from the *BFC* of the *IB* under heavy loads. Conversely, the *BFC* of the *IB* plays the role of the permanent initiator. In addition, the responder function suspends the access to passing *FG* slots in the scheduling phase of buffer cycles. This helps to avoid buffer overflow.

Due to the finite property of buffer capacity, the algorithm of fairness threshold computation used in the initiator algorithm of BfCP is different from that of BwCP. To compute the values of *FT* and *H*, the algorithm in the following is described by using an *OB* as an example. As for an *IB*, the algorithm can be applied as well. Since the inter-traffic flows come simultaneously from several source bridges, balancing the utilization of the buffer capacity among different transmission requests is necessary to keep the buffer capacity to a sufficient level. To fairly share the buffer capacity, we adopt a discipline called *less buffer's occupation first (LBOF)* to compute the *FT* value. This discipline is very intuitive. It prioritizes buffer access to those source bridges that may be excluded from utilizing the buffer capacity. To demonstrate the algorithm, the following assumptions are made. The *OB* of all bridges has the size of *C* slots. The capacity of the *OB* of bridge  $B_k$  is insufficient so that a buffer cycle is started.  $A_k$  is the occupied buffer capacity while the *RSV* coming back to the initiator. Note that  $A_k < C$ ; otherwise, the buffer overflow occurs. In the algorithm, *S* stands for the set of all bridges and *m* is a counter for the number of removed bridges from *S*.

#### Buffer *FT* computation algorithm.

```

m = 0;
S = {B1, B2, ..., Bk-1, Bk+1, ..., Bn};
while (true) {
 Topt = ⌊ $\frac{(C - A_k) + \sum TC_i^k \forall B_i \in S}{n - 1 - m}$ ⌋;
 EndOfLoop=true;
 for (i = 1, (i ≤ n) ∧ (i ≠ k), i = i + 1) {
 if ((Topt - TCik) ≤ 0) ∧ (Bi ∈ S) {
 S = S - {Bi};
 m = m + 1;
 EndOfLoop=false;
 }
 }
 if (EndOfLoop=true)
 break;
}
FT = Topt; H = 0;
for (i = 1, (i ≤ n) ∧ (i ≠ k), i = i + 1) {
 if (Topt > TCik) {
 if ((Topt - TCik) > RCik)
 H = H + RCik;
 else H = H + (Topt - TCik);
 }
}
return(FT, H);

```

/\*m counts the number of bridges removed\*/  
/\*the set of all bridges except B<sub>k</sub>\*/  
/\*compute the threshold\*/  
/\*a test flag for ending the while loop\*/  
/\*this is the set operation to remove B<sub>i</sub>\*/  
/\*increase one removed bridge\*/  
/\*leave the while loop\*/  
/\*calculate the number of reserved slots\*/

In the first iteration of the algorithm, the value of  $Topt$  is computed. It is used to identify the bridges (say  $B_i, i \neq k$ ) that have transmitted more slots to  $B_k$ , i.e.,  $(Topt - TC_i^k) \leq 0$ . These bridges are then removed from  $S$ . Subsequently,  $Topt$  is computed again by considering the remaining bridges in  $S$  and the remaining size of the destination  $OB$ , namely,  $(C - A_k)$  slots. The iteration is repeated until it stops under the condition  $(Topt - TC_i^k) > 0$  for all remaining nodes in  $S$ . At this point, the  $FT$  value is determined. Additionally,  $H$  is also computed at the same time. The values of  $FT$  and  $H$  are then returned to the initiator algorithm. As the values have been computed, it implies that both the avoidance of buffer overflow and the fair usage of buffer capacity can be achieved.

To illustrate the determination of  $FT$  and  $H$  using the algorithm and the occurrence of buffer cycles, we take the scenario given in Fig. 4 as an example again. Assume that  $B_2, B_3$ , and  $B_4$  have the inter-traffic flows to the  $OB$  of  $B_1$  and the capacity of this buffer is insufficient. As shown in Fig. 6(a), the  $BFC$  for the  $OB$  of  $B_1$  switches to the active state and issues a RSV to collect the parameter pairs  $(TC_i^1, RC_i^1)$ , where  $i = 2, 3$  or  $4$ . In Fig. 6(b), the scheduling process and the values of parameters are sketched. In the first  $FT$  computing iteration,  $Topt = 7$  and  $B_3$  is removed from  $S$ . In the second iteration,  $Topt = 6$  and all conditions are satisfied. Then, the  $FT$  value is 6 and the number of slots reserved ( $H$ ) is 4. At this point,  $B_1$  issues a CFM with  $FT = 6$  and marks four FG slots as FR slots successively (Fig. 6(a)). Finally,  $B_2$  uses three FR slots and  $B_4$  uses the last FR slot for their inter-traffic flows to the  $OB$  of  $B_1$ .

### 3.4. Coordination of multiple reservation cycles

Occasionally, bandwidth and buffer cycles may be started simultaneously and overlap each other. For example, as shown in Fig. 4, suppose that Local ring 1 and the  $IB$  of  $B_1$  are heavily loaded at the same time. As soon as  $U_{1,3}$  has inter-traffic to the  $IB$  of  $B_1$ , it has to face a bandwidth and a buffer cycle. If the occurrence of the bandwidth cycle proceeds the buffer cycle,  $U_{1,3}$  has to suspend the access of free slots and

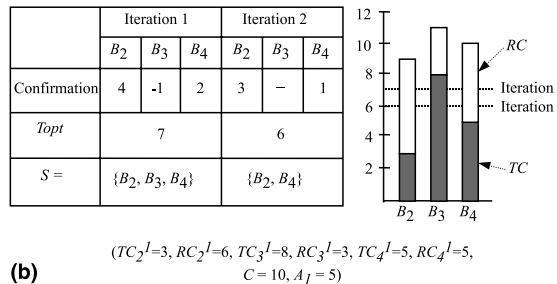
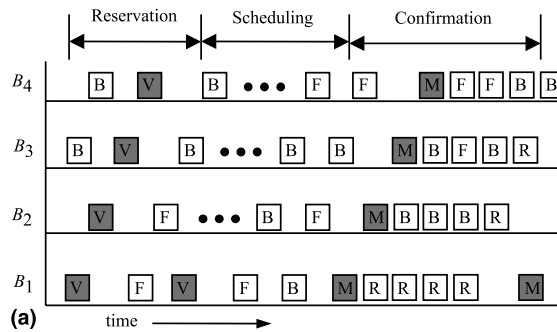


Fig. 6. (a) A buffer cycle; (b) a scheduling computation for buffer capacity.

change to the buffer cycle. Otherwise, the excessive traffic may overrun the *IB* of  $B_1$ . Alternatively, when the buffer cycle gets precedence, there are three possibilities. For the first case, if  $U_{1,3}$  determines the deferments for the buffer cycle, then it has to ignore the subsequent bandwidth cycle and stay in the buffer cycle in order to avoid the possible buffer overflow. For the second case, if  $U_{1,3}$  determines the confirmations for both the buffer and the bandwidth cycle, then the smaller quantity of confirmations is preferred so as to maintain the fair bandwidth access as well as avoid the buffer overflow. For the last case, if  $U_{1,3}$  determines the confirmation for the buffer cycle and the deferments subsequently for the bandwidth cycle, then it has to throttle the access of free slots to the *IB* immediately in order to maintain the fair access of bandwidth. To a greater extent, the same control approach can be applied to bandwidth and buffer cycles occurred simultaneously in the backbone ring.

### 3.5. Resolution of head-of-line blocking

In general, the *IB* of a bridge may store the packets belonging to several inter-traffic flows destined for different destination bridges. If the capacity of the *OB* of one of the destination bridges is insufficient, the inter-traffic flow destined for that bridge would block all succeeding inter-traffic flows until the marked FR slots arrive. More seriously, if the capacity of the *OB* of two or more destination bridges were insufficient, a *head-of-line blocking* phenomenon would happen due to waiting for different sequences of marked FR slots. For example, suppose that there are two inter-traffic flows in an alternate sequence toward  $B_1$  and  $B_3$  in the *IB* of  $B_2$  as shown in Fig. 4. If the reservation cycle for the *OB* of  $B_1$  is started, the traffic flow to  $B_1$  will block the flow to  $B_3$  until the FR slots marked by  $B_1$  arrive. Again, suppose that the cycle for the *OB* of  $B_1$ , and  $B_3$  are started simultaneously. A more serious blocking will happen in the *IB* of  $B_2$ . Actually, the frequent occurrence of head-of-line blocking is much harmful to the performance of the global network.

To tackle this issue, a scheme using a multiple linked list structure to hold multiple traffic flows in an *IB* is suggested. With some software methods, this scheme constructs a multiple linked list structure where every linked list in the structure is used to hold a subqueue of packets destined for one specific bridge. Simultaneously, the global FIFO order for packet arrival is also retained. When FG slots pass through, the transmission of a packet is selected from those subqueues in the global FIFO order. However, for the FR slots corresponding to one specific reservation cycle passing, the corresponding subqueue is selected and the data packet is transmitted. For example, in Fig. 4, two subqueues storing the inter-traffic flows to  $B_1$  and  $B_4$  in the *IB* of  $B_3$  are shown. To implement such a scheme, the approach of using high performance microprocessors and random access memory is feasible due to the rapid development of microelectronics technology. Since it is an engineering issue to the implementation of a multiple subqueue scheme, we skip the detailed explanations.

## 4. Simulation models

The simulation models include a network model, a user node model, a bridge model, and load generation models. For the network model, it is based on the single hierarchical ring with all nodes placed equidistantly. In accordance with a standardized transmission line, namely, STM-8, the transmission rate of local rings is set to 1.2 Gbit/s. For the backbone ring, the rate is set to 2.5 Gbit/s (STM-16). The transmission medium is assumed to be the optical fiber with 5  $\mu$ s/km propagation delay. The node latency, which occurs due to storing and checking the header of slots, associated with bridges or user nodes is set to 1 slot (72 octets). The error rate of the transmission medium is not considered. For the user node and the bridge model, the node structures have been sketched in Fig. 3. To cope with simulations, each *BWC* or *BFC* is simplified to be as a command delay buffer and a set of registers used to count the slots transmitted, slot requests, confirmations, and deferments. Actually, each user node is modeled by a set of a transmitter,

receiver, insertion buffer, transmission buffer, *BWC*, and *BFC*; each bridge by an *IB* and *OB* and two sets of a transmitter, receiver, insertion buffer, *BWC*, and *BFC*.

In conjunction with the models described above, the following load models are used in our simulation experiments. The burst–silence load model is used to analyze the relationship between network throughput and averaged transfer delay. Assume that the number of slots in each burst has a shifted geometric distribution with a mean value of 10 slots, and the duration of a silence period is determined by an exponential distribution. This model is selected because the length of silence can be varied to control the density of traffic with respect to the offered load. The heavy load model is used in the performance simulation with respect to the network size and the number of user nodes supported. This model is implemented by always having a new packet to send in the transmission buffer. Another important factor that can affect the simulation is the traffic distributions. Suppose that the user nodes connected by different local rings are treated as constituting different communicating communities. The traffic distribution resulting from the interacting communities is more likely than the uniform traffic in a network with a large number of users. To characterize the locality property, the amount of intra-traffic over the total amount of inter- and intra-traffic is defined as the ratio for the traffic distribution with locality. We will compare the performance of the single hierarchical ring according to different locality ratios.

To investigate the performance characteristics of CRMA-HR, the network throughput, access delay, and averaged total delay are chosen for the simulations. The network throughput is defined as the total number of packets received by all user nodes divided by the number of packets through a link in the network within one second. The access delay is defined as the time between the arrival of a packet at the head of the transmission buffer until the first bit is transmitted. The transfer delay of a packet is defined as the time between the transmission of the first bit from the source until the last bit is received by the destination. The total delay of a packet is defined as the sum of the access and the transfer delay. Hence, the averaged total delay is given by averaging over all packets generated during the simulation time. To perform the simulations, the parameters used are summarized as follows.

|                                |                                                     |
|--------------------------------|-----------------------------------------------------|
| Configuration                  | single hierarchical ring                            |
| Number of nodes                | 50–250                                              |
| Network length                 | 70–1400 km                                          |
| Transmission rate              | backbone (2.5 Gbit/s), local rings (1.2 Gbit/s)     |
| Size of insertion buffers      | one packet of maximal size                          |
| Size of transmission buffers   | 100 slots                                           |
| Size of input/output buffers   | 2000 slots                                          |
| Delay of insertion buffer      | 1 slot                                              |
| Propagation delay              | 5 $\mu$ s/km                                        |
| Slot length                    | 72 octets                                           |
| Scheduling latency (bandwidth) | local rings (10 $\mu$ s), backbone ring (5 $\mu$ s) |
| Scheduling latency (buffer)    | input buffer/output buffer (5 $\mu$ s)              |
| Load models                    | burst–silence/heavy                                 |

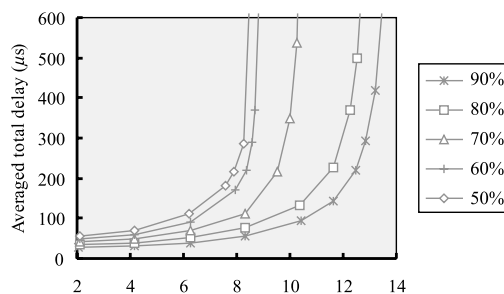
In the following, the simulations are based on the simulation software prepared using SIMSCRIPT II, the simulation experiments are replicated corresponding to the variance reduction technique with different sequences of pseudo-random numbers and the results are obtained with 95% confidence levels.

## 5. Performance

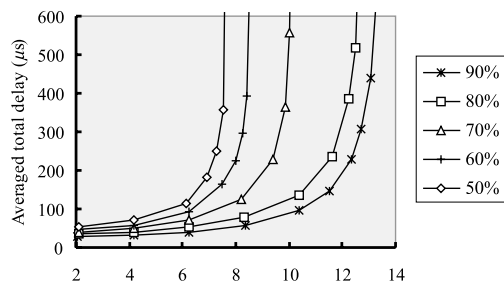
In the first simulation, the network configuration consists of 10 local rings (with five user nodes and one bridge in each local ring) and a 70-km network with all nodes placed equidistantly. To analyze the per-

formance properties of CRMA-HR, the balanced and the unbalanced traffic pattern are considered. For the balanced pattern, all nodes are assumed to send intra-traffic to all destinations uniformly, except to themselves, within a local ring. As for inter-traffic, the same assumption applies the uniform distribution to all user nodes within the single hierarchical ring, except to those nodes in the same local ring with the source node. For the unbalanced pattern, assume that inter-traffic from each bridge is sent to all bridges uniformly, except that the inter-traffic from the first bridge is only sent to the fifth bridge. As for intra-traffic, the same assumption is made as the balanced case. To perform the simulations for this network configuration, the locality ratios of 50%, 60%, 70%, 80%, and 90% are chosen. Also, the throughputs of a local ring, the backbone ring, and the global ring are normalized with respect to the link capacity of local rings, i.e., 1.2 Gbit/s, for the purpose of comparison.

The input load is based on the burst–silence model. Fig. 7 shows the growth of averaged total delay with respect to global throughput for the balanced and the unbalanced pattern according to the five locality ratios. The asymptotes of the lower locality ratios show the lower throughput for both the balanced and the unbalanced pattern and vice versa. To see this situation, consider two extreme cases: the 0% and the 100% locality ratio. For the 0%, there is no intra-traffic according to the definition. All of the traffic transferred to their destinations need be switched through the backbone ring. Thus, the backbone ring would saturate very quick due to lots of inter-traffic. At this point, the global throughput is worst. For the 100%, the traffic flows are all the intra-traffic transferred to their corresponding destinations inside the respective local ring. The global throughput becomes the summation of the network throughput of all local rings. Actually, the balanced and the unbalanced pattern are the same in the case of the 100% owing to no inter-traffic. However, as the locality ratio decreasing, the performance difference between them is gradually distinct as shown in Fig. 7(a) and (b). As can be seen, the first bridge that always transmits packets to a fixed destination bridge makes part of the backbone ring congested. Thus, for the unbalanced pattern, the



(a) Global throughput



(b) Global throughput

Fig. 7. Averaged total delay vs. global throughput for (a) the balanced pattern and (b) the unbalanced pattern.

performance is worse gradually as the locality ratio decreased. From another point of view, it can be seen that the backbone ring dominates growth of the global throughput. If the bandwidth of the backbone ring is enlarged, more inter-traffic can be switched for the case of the lower locality ratios (e.g., 50–70%). This benefits the global throughput.

For the following simulations, the input load is based on the heavy load model. In Fig. 8, the curves of the simulated throughput for the backbone ring, local rings, and single hierarchical ring are depicted respectively according to the locality ratios, where the throughput of local rings is an average over all local rings. Owing to the different scales, the throughputs of the backbone ring and local rings are sketched with respect to the left vertical axis, and the global throughput of the single hierarchical ring is sketched with respect to the right vertical axis. From observing Fig. 8(a) and (b) for the balanced and the unbalanced traffic pattern, two attractive points can be found. First, in comparison with the throughput results for the balanced pattern, the throughputs of the backbone ring, local rings, and single hierarchical ring for the unbalanced pattern degrade gradually as the locality ratio decreased. The reason is that the inter-traffic flowing through the network links from the first bridge to the fifth bridge are blocked by the unbalanced traffic. Especially, it is even worse for the case of the lower locality ratios. Second, it is interesting to see that the multiplication of the averaged throughput of local rings and the number of local rings minus the throughput of the backbone ring is very approximate to the global throughput regardless of the traffic patterns. Concisely, the situation is twofold: the throughput of the backbone ring is produced by the inter-traffic from all local rings, and on the other hand, the throughput of a local ring includes part of the inter-traffic received by the bridge of the local ring. In that case, the duplicated count for the throughput in the backbone ring and local rings should be subtracted from the multiplication of the averaged throughput of local rings and the number of local rings to give the exact global throughput.

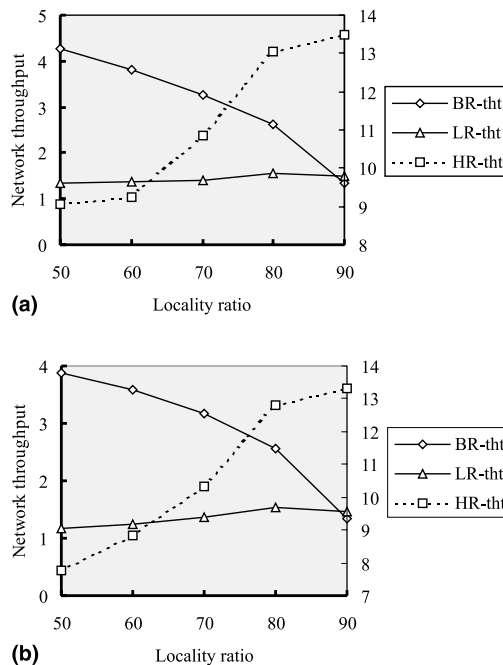


Fig. 8. Network throughput vs. locality ratio for (a) the balanced traffic and (b) the unbalanced traffic (BR-tht: backbone ring's throughput; LR-tht: local ring's throughput; HR-tht: hierarchical ring's throughput).



A MAC protocol had better be independent of network size. This item is related to the well-known *a-parameter* [3]. Any protocol that is sensitive to this parameter will limit its efficiency and usefulness to the larger-size networks. To evaluate the sensitivity of CRMA-HR, the global throughput and access delay with respect to the network size for different locality ratios are simulated and plotted in Fig. 9. In this simulation, the simulation conditions are the same as the above except that the network length between any two nodes is extended from 1 to 20 km (i.e., the network size from 70 to 1400 km). As can be seen, the global throughput for each locality ratio shrinks slightly with respect to the increase of network size drawn in Fig. 9(a). Especially, the global throughput of low locality (e.g., 50% or 60%) shrinks more than the throughput for high locality. This is due to the traffic congestion occurred in the backbone ring in case of low locality ratios. As for the averaged access delay with respect to the network length drawn in Fig. 9(b), the delay for each locality ratio varies slightly as well. Nevertheless, the delay increases as the increase of the locality ratio except for 90%. Note that there is a conversion phenomenon between 90% and 80% in Fig. 9(b). To see this, we can find that the throughput of local rings for the 80% locality ratio is higher than the throughput for 90% depicted in Fig. 8(a). It can be seen that the traffic for 80% is denser than for 90% in local rings. In that case, the denser traffic makes the access delay of 80% be longer.

For a large network, the MAC protocol must be efficient to support a great number of attachments. To investigate the performance property of CRMA-HR with respect to the number of user nodes, two network configurations are considered. Configuration 1 has 10 bridges and the number of user nodes ranges from 5 to 25. On the contrary, Configuration 2 has 10 user nodes in each local ring and the number of bridges is from 5 to 25. That is, the total number of user nodes ranges from 50 to 250. Alternatively, due to the nature of locality ratio to the hierarchical ring networks, the ratios of 60% and 80% are chosen for the demonstration of performance impact. Fig. 10 shows the global throughput and access delay with respect to the

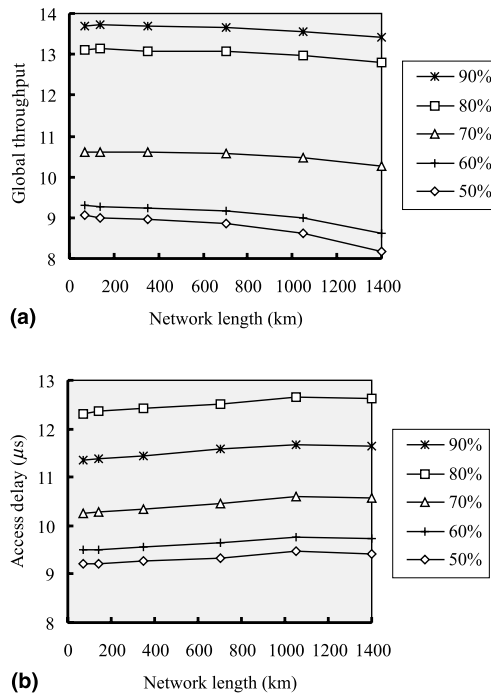


Fig. 9. (a) Global throughput vs. network length; (b) access delay vs. network length.

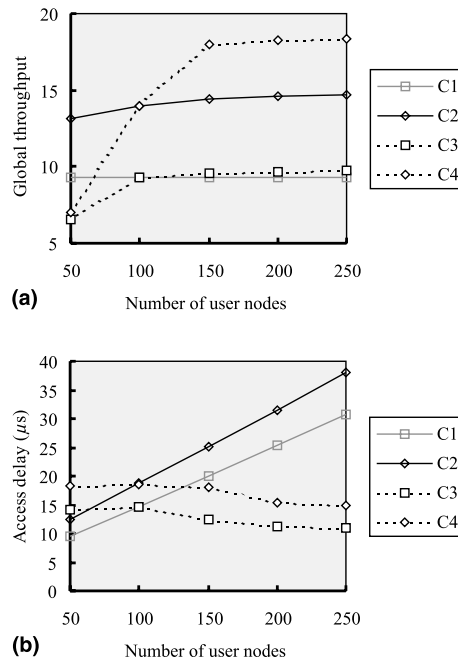


Fig. 10. (a) Global throughput vs. number of user nodes; (b) access delay vs. number of user nodes (C1: Configuration 1 with the locality ratio 60%; C2: Configuration 1 with the locality ratio 80%; C3: Configuration 2 with the locality ratio 60%; C4: Configuration 2 with the locality ratio 80%).

node number. For the combinations of two configurations and two locality ratios, four curves (denoted as C1, C2, C3, and C4) are plotted in Fig. 10, respectively.

To analyze the impact to the global throughput in Fig. 10(a), two types of comparison are made according to configurations and locality ratios. For the configurations, it can be seen that the global throughput for Configuration 2 (curves C3 and C4) is better than for Configuration 1 (curves C1 and C2) as the node number exceeds 100. Since the number of local rings is increasing in Configuration 2, it is beneficial for the growth of the global throughput until the backbone bandwidth is fully occupied. As for Configuration 1, the number of local rings is fixed so that the increase in throughput is bounded. Whether the backbone ring or the local rings may be saturated, the throughput limitation will be approached rapidly. For the locality ratios, it is obvious that curves C2 and C4 (80%) give higher throughputs than C1 and C3 (60%). In the simulations, it is due to the backbone ring saturated by the heavier inter-traffic in the locality of 60%. Additionally, under the saturation of the backbone ring, some interesting points can be found from curves C1, C3, and C4. For C1 and C3, the global throughput is almost unrelated to the number of nodes. Next, the saturation of the backbone ring starts from 100 nodes for C3 and 150 nodes for C4. For the impact of the access delay in Fig. 10(b), the comparison is as above according to configurations and locality ratios. Explicitly, it can be seen that the access delays are linear to the number of nodes for Configuration 1 (C1 and C2), but non-linear for Configuration 2 (C3 and C4). In this situation, the explanation can be given from the viewpoints of access opportunities and the number of nodes. In Configuration 1, the relative access opportunity of each node to a local ring decreases by degree with respect to the increasing node number. Therefore, the access delays by C1 and C2 increase linearly. Alternatively, since the node number of local rings is fixed in Configuration 2, the access opportunity of a user node varies slightly. As the number of local rings increases, the shared throughput and the traffic density of each local

ring reduce so that the access delay can be shorter. Hence, it makes the access delays by C3 and C4 change in slightly decrease according to the node number. As for the effect of the locality ratio to the access delay, it is shown that the higher ratio has longer delay owing to the denser intra-traffic and fewer access opportunities in local rings.

## 6. Conclusions

In this paper, a large-scale Gbit/s MAN based on the hierarchical ring topologies using CRMA-HR has been presented. The main objectives of the network are to support a great number of users with highly aggregated bandwidth and to cover a large geographical area than traditional MANs. For CRMA-HR, it consists of BwCP and BfCP. The BwCP protocol controls the utilization of network bandwidth of the subnetworks under heavy or unbalanced workloads. While incorporating with a simple busy/free mechanism, the high network utilization, enforced fairness, and bounded access delay can be achieved. The BfCP protocol controls the utilization of buffer capacity of the bridge buffers (namely, *IB* and *OB*). Without this protocol, the network congestion due to buffer overflow can block inter-traffic to cause the performance degradation of the global network; the average transfer delay becomes unpredictable. In particular, these control protocols adopt the same cyclic reservation scheme, which can simplify the development of network interface units and reduce the implementation cost. It is inevitable that CRMA-HR has to face with different network variations, e.g., multiple-level hierarchical rings, double hierarchical rings, and so forth. Fortunately, CRMA-HR fits these network variations well. For example, extending to multiple-level networks, an inherent advantage of the hierarchical ring configuration is its self-similarity. Whatever the number of levels may be, the network resources used to facilitate the communications in hierarchical rings are the network bandwidth and buffer capacity. Hence, CRMA-HR can work in a multiple-level hierarchical ring with the aforementioned address scheme. In addition, each subnetwork may be a dual-ring, which can be viewed as two counter-rotating single rings. Obviously, CRMA-HR can be applied to each single ring in the dual-ring subnetworks of a double hierarchical ring.

To investigate the performance properties of this network, several simulations were designed, implemented, and analyzed. From the simulation results, it is shown that this network has excellent global throughput and average transfer delay according to the locality of the traffic distribution. Especially, its related performance characteristics can fit to the network scalability in terms of the network size and the number of attachments.

In summary, our approach for such a large-scale Gbit/s MAN has the properties of simple routing, straightforward media access, high connectivity for attachments, and self-similarity of the network topology for extension. Next, CRMA-HR can satisfy the fairness requirement and operate independent of the network size and the number of attachments. Especially, to reduce the complexity of MAC protocol, a generic cyclic reservation-based mechanism is developed to facilitate the access control for different network resources. Finally, some work is unfinished, for instance, the optimization of arranging multiple reservation cycles and other approaches for resolving the head-of-line blocking problem.

## References

- [1] M. Ajmone Marsan, G. Albertengo, C. Casetti, F. Neri, G. Panizzardi, On the performance of topologies and access protocols for high-speed LANs and MANs, *Comput. Networks and ISDN Syst.* 26 (1994) 873–893.
- [2] H.R. van As, Media access techniques: the evolution towards terabit/s LANs and MANs, *Comput. Networks and ISDN Syst.* 26 (1994) 603–656.
- [3] H.R. van As, W.W. Lemppenau, H.R. Schindler, P. Zafiropulo, CRMA-II: a MAC protocol for ring-based Gb/s LANs and MANs, *Comput. Networks and ISDN Syst.* 26 (1994) 831–840.

- [4] H.R. van As, W.W. Lemppenau, P. Zafropulo, Performance of CRMA-II: a reservation-based fair media access protocol for Gbit/s LANs and MANs with buffer insertion, in: Proceedings of the EFOC/LAN'92, Paris, France, June 1992, pp. 162–169.
- [5] J. Brassil, A.K. Choudhury, N.F. Maxemchuk, The Manhattan Street Network: a high performance, highly reliable metropolitan area network, *Comput. Networks and ISDN Syst.* 26 (1994) 841–858.
- [6] C. Douligeris, L.N. Kumar, Fairness issues in the networking environment, *Comput. Comm.* 18 (4) (1995) 288–299.
- [7] M. Gerla, L. Kleinrock, Congestion control in interconnected LANs, *IEEE Network* 2 (1) (1988) 72–76.
- [8] K. Imai, T. Ito, H. Kasahara, N. Morita, ATMR: asynchronous transfer mode ring protocol, *Comput. Networks and ISDN Syst.* 26 (1994) 785–798.
- [9] W.T. Lee, L.Y. Kung, The optimized architecture of hierarchical ring networks, in: Proceedings of the IEEE SICON'97, Singapore, April 1997, pp. 247–259.
- [10] W.W. Lemppenau, H.R. van As, H.R. Schindler, A 2.4 Gbit/s ATM implementation of the CRMA-II dual-ring LAN and MAN, in: Proceedings of the EFOC/LAN'93, The Hague, Netherlands, June 1993, pp. 274–281.
- [11] N.F. Maxemchuk, The Manhattan Street Network, in: Proceedings of the IEEE GLOBCOM'85, 2–5 December 1985, pp. 255–261.
- [12] N.F. Maxemchuk, A comparison of linear and mesh topologies – DQDB and the Manhattan Street Network, *IEEE J. Selected Areas Comm.* 11 (8, October) (1993) 1278–1289.
- [13] N.F. Maxemchuk, Problems arising from deflection routing, live-lock, lockout, congestion and message reassembly, in: NATO Workshop on Architecture and Performance Issues of High Capacity Local and Metropolitan Area Networks, France, June 1990, pp. 209–233.
- [14] T. Meuser, Performance comparison of media access protocols for Gbit/s networks in the local area, *Comput. Comm.* 18 (1, January) (1995) 4–14.
- [15] M.M. Nassehi, CRMA: an access scheme for high speed LANs and MANs, in: Proceedings of the IEEE ICC'90, Atlanta, GA, 16–19 April 1990, pp. 1697–1702.
- [16] Y. Ofek, Overview of the MetaRing architecture, *Comput. Networks and ISDN Syst.* 26 (1994) 817–829.
- [17] F.E. Ross, An overview of FDDI: Fiber Distributed Data Interface, *IEEE J. Selected Area Comm.* 7 (1989) 1043–1051.
- [18] IEEE Standards for Local and Metropolitan area networks: Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN), 802.6, 1990.



**Wen-Fong Wang** was born in Taiwan and received his Ph.D. degree in Electrical Engineering in 1998 from National Cheng-Kung University, Taiwan, ROC. He has been a researcher of Telecomm Lab., Chunghwa Telecomm Co., Ltd., since 1989. His research interests include Performance Evaluation, Communication Protocol Design, High-Speed Networks, and Protocol Engineering.



**Wen-Shyang Hwang** received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from National Cheng-Kung University, ROC, in 1984, 1990 and 1996. He is an associate professor of Technology, Taiwan, ROC. His current research interests are in Multi-channel WDM Networks, Performance Evaluation, Qos, RSVP WWW Database Applications.



**Jun-Yao Wang** received the B.S. degree in Computer Science & Information Engineering from Tatung University in 1992. In 1994, he received the M.S. degree in Electrical Engineering from National Cheng-Kung University. He is also a Ph.D. candidate. His current interests include High-Speed Multi-access Protocol Local Area Networks, and Performance Evaluation.