PERGAMON

Computers and Electrical Engineering 27 (2001) 173–199

Computers and Electrical Engineering

www.elsevier.com/locate/compeleceng

On the resource reservation approach to design a large-scale and ultra high-speed MAN

Wen-Fong Wang[a,*], Jun-Yao Wang[a], Wen-Shyang Hwang[b]

[a]*Computer Laboratory, Department of Electrical Engineering, National Cheng-Kung University, Tainan 701, Taiwan, ROC*

[b]*Department of Electrical Engineering, National Kaohsiung Institute of Technology, Kaohsiung 807, Taiwan, ROC*

Received 25 March 1998; accepted 27 January 1999

Abstract

To support a great number of dispersed users in a wider area with high-speed communication services, we develop a large-scale and ultra high-speed MAN based on hierarchical ring configuration. The network is constituted by backbone and local rings, which are connected by bridges. In such a network, traffic congestion may always occur due to the mismatch of transmission speed between backbone and local rings. To cope with the issue, we adopt a resource reservation approach based on the method of cyclic reservation-based access control for controlling access to different network resources, viz. network bandwidth and bridge buffers. By this approach, a MAC protocol that can achieve fair access to network resources and avoid traffic congestion is proposed for the network. To evaluate the performance properties of the network, several simulation experiments are performed and some optimistic results are revealed.

Keywords: LAN; MAN; Reservation mechanism; Hierarchical ring networks; High-speed MAC protocol

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 communication capability. Today, the popular MANs are

* Corresponding author. Kaohsiung P.O. Box 742, Kaohsiung city, Taiwan, ROC.
E-mail address: wwf@ms.chttl.com.tw (W.-F. Wang).

0045-7906/01/$ - see front matter
PII: S0045-7906(00)00012-4

built with the standards of Fibre Distributed Data Interface (FDDI) and Distributed Queue Dual Bus (DQDB) [17,18]. They are built with modest, existing technology and linear topology, and operate at the speed of hundred or a few hundred megabits per second. Recently, as 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 networking services. Thus, those existing networks are inefficient to cope with the user requirements. The trend is to develop larger size and higher throughput MANs.

With the advent of optical fiber as a medium, ultra high-speed communications about gigabit or even terabit are feasible. In this study, it is assumed that optical fiber is the basic transmission medium and Gbit/s the basic rate. To be a large-scale network, the condition of rich space diversity is necessary [1,2]. For example, a remarkable network presented in the literature is grid network, e.g., the Manhattan Street Network (MS_Net) [11,12]. It has rich connectivity and high aggregate bandwidth. However, it suffers from a variety of problems [5,13]. As a result, its service quality is affected significantly. To seek another solution, a hierarchical ring network (see Fig. 1) is investigated by considering multiple connectivity. It is composed of a number of local rings, which are in term connected by a backbone ring with bridges. Based on slotted and buffer insertion access mechanisms [2], 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 has the potentiality extending to multiple-

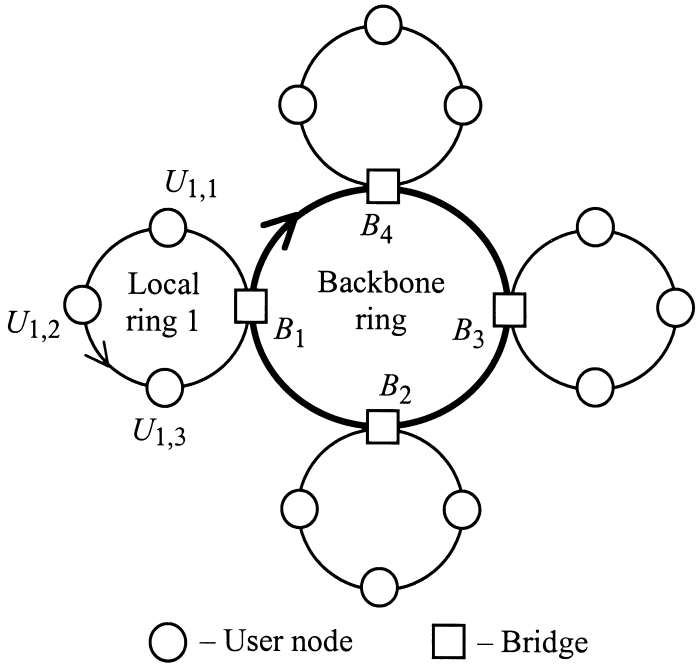


Fig. 1. The topology of a 4×4 single hierarchical ring.

level hierarchical configuration [9]. Apparently, this type of network fits to cover larger area and supports more users than traditional MANs.

In a hierarchical ring network, bridges are necessary to switch traffic to/from some other local ring (called inter-traffic) or within the same local ring (called intra-traffic). In such a network, traffic congestion always happens [7]. It incurs owing to the difference of network bandwidth between the higher speed backbone ring and the lower speed local rings, and also owing to the unbalanced workload among different inter-traffic flows. For instance, if an inter-traffic flow becomes heavy, a long queue of packets is built up in the destination bridge that this inter-traffic flows through and some packets are eventually lost due to buffer overflow. Considerable congestion also arises when packets coming from different bridges gather on a certain bridge simultaneously. From these facts, buffer capacity is as important as network bandwidth in a hierarchical ring network. Absolutely, we can treat them as two different network resources since both of them are finite and necessary for communications among users. To resolve the congestion issue, the control mechanisms for buffer capacity and bandwidth should be integrated simultaneously into a MAC protocol for the hierarchical ring network.

To facilitate access control to network resources, we adopt a resource reservation approach based on the method of cyclic reservation-based access control scheme [14]. Note that the scheme was originally proposed by IBM, and had developed two versions, i.e. Cyclic Reservation Multiple Access (CRMA) [15] and CRMA-II (version two of CRMA) [3,4,10]. In the literature, it had been shown to be rather efficient in comparison to other MAC schemes for high-speed LANs and MANs [1,8,14,16]. In this study, a protocol named as the Cyclic Reservation Multiple Access for Hierarchical Rings (CRMA-HR) is proposed for hierarchical ring networks. This protocol can be used to arbitrate utilization of the network resources from numerous access requests. To cope with different resources, the CRMA-HR protocol is divided into two parts: the bandwidth control protocol (BwCP) and the buffer control protocol (BfCP). Basically, each node (a user node or bridge) has to play the role of either initiator or responder while executing the control protocols. As a node starves for bandwidth of the network, it performs BwCP to start a series of reservation cycles to coordinate fair access to bandwidth until the starvation ends. At this time, the node plays the role of initiator. Other nodes (responders) receiving the coordination message from the initiator join the activities of reservation cycles to share bandwidth fairly. For BfCP, it is executed when a bridge finds that the capacity of one of its buffers is insufficient. If inter-traffic flows through the bridge starve for buffer capacity, the bridge (initiator) starts a series of reservation cycles to prevent possible network congestion. Other bridges (responders) that receive the coordination message and have inter-traffic to the initiator join the activities of reservation cycles to share the remaining capacity of the target buffer until the starvation ends. As for the solution to sharing network resources among nodes fairly, a few fairness threshold computation algorithms may be employed to determine fair shares. However, the algorithms for BwCP and BfCP are different due to various resource types.

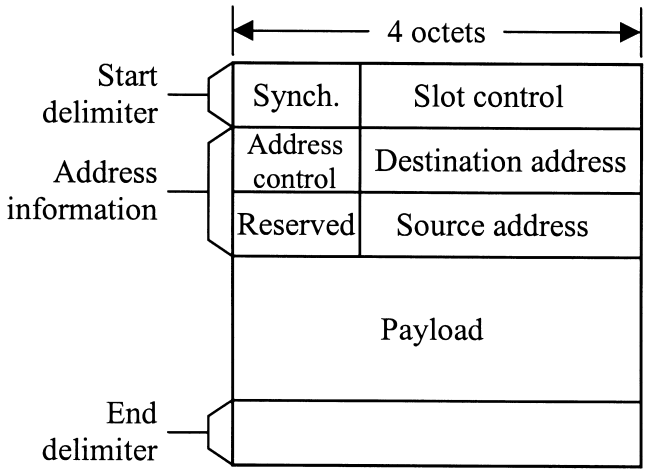
Note that both the control protocols adopt the reservation mechanism. As a result, most of the components in the network interface unit for a bridge or user node can be shared so that the implementation complexity of CRMA-HR is reduced. Based on optical fibers with data rate up to Gbit/s, the model of a hierarchical ring network and the basic access mechanisms

are briefly sketched in the next section. Section 3 describes the two parts of CRMA-HR for network bandwidth and buffers. Particularly, there are a number of issues such as coordination of multiple reservation cycles of different resources, head-of-line blocking, and variations in hierarchical ring configuration. The resolutions are also demonstrated. To investigate the performance of CRMA-HR, Section 4 carries out several simulation experiments under different traffic conditions. The final section presents the conclusions.

2. A hierarchical ring network

A single hierarchical ring network (see Fig. 1) consists of a number of local rings, which connect a lot of user nodes. Local rings are then connected to a higher bandwidth backbone ring via bridges. Each component ring (backbone or local ring) can be a single or dual ring. Thus, other variations may exist, namely dual backbone ring connecting single local rings, all component rings being dual rings, and so on. For simplicity, the single hierarchical ring is considered first.

For the purpose of data transmission, a slotted transmission structure is adopted with the slot being the unit of reservation. In Fig. 2, a slot format is shown. It consists of a



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.

start and end delimiter pair with address information and payload embedded [3]. Start delimiter and address information forms the header of a slot. For the start delimiter, it consists of a synchronization part and a slot control part, which contains Slot_type, Reservation_cycle, Status, Priority, Monitor, and Concatenation. The Slot_type field identifies a transmission entity as a slot, as a multi-slot, or as one of command slots. The Reservation_cycle field indicates none, cycles for bandwidth, or 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 (denoted as FG) and then mark them as the Busy-Gratis slots (denoted as BG) while using. If a slot is reserved in a reservation cycle, it is marked as a Free-Reserved slot (denoted as FR). The Concatenation field specifies whether a slot is single (e.g., an ATM cell) or is part of a multi-slot frame. In the latter case, its position is further defined as first, middle, or last. The Monitor field is used for detecting and freeing the circulated slots in a component ring.

In hierarchical rings, the address scheme is much more complex than the scheme for single or dual rings. It consists of the parts of address control, destination address, and source address (see Fig. 2) to provision larger address space for hierarchical rings. Although the global address up to eight octets in each slot wastes a little extra network bandwidth, it can support the addressing of a large number of network nodes and is necessary for large size networks. The address control part contains Hierarchical_level and Address_type. The Hierarchical_level field indicates the level number of network hierarchy. 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 the hierarchical level two bits long and the address type six bits long, the destination and source addresses with the field length of 3 octets (or 24 bits) can be separately divided into six 4 bits groups. If the level number is two and the address type is the pattern of 000111, then the first three groups are the address for bridges in backbone ring and the last three groups are the address for user nodes in local rings. Therefore, in hexadecimal format, a bridge address 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, supposing that the level number is three and the address type is the pattern of 110011, then the first two groups are the address for bridges in level 2 backbone ring, the second two groups are the address for bridges in level 1 backbone rings, and the last two groups are the address for user nodes in local rings. In this way, the address scheme can be applied to many different address types and multiple-level hierarchical rings.

A user node (see Fig. 3(a)) is used to connect customer 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: passive and active. On the passive state, it monitors utilization of transmission medium. When one or more FG slots arrive to a user node and its insertion buffer is empty, media access for the packet in the head of the transmission buffer is taken. To allow transmission of a complete packet in each node, delaying incoming slots in the insertion buffer supports transmission of a multi-slot packet. Because of destination release strategy, slots may be used several times by spatial reuse. If a user node suffers

poor access to the network, the BWC of the node switches to the active state and plays the role of initiator. At this point, the BWC of others switches to the active state as well and plays the role of responder. BFC is described subsequently.

A bridge (see 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 BWCs for the local ring and

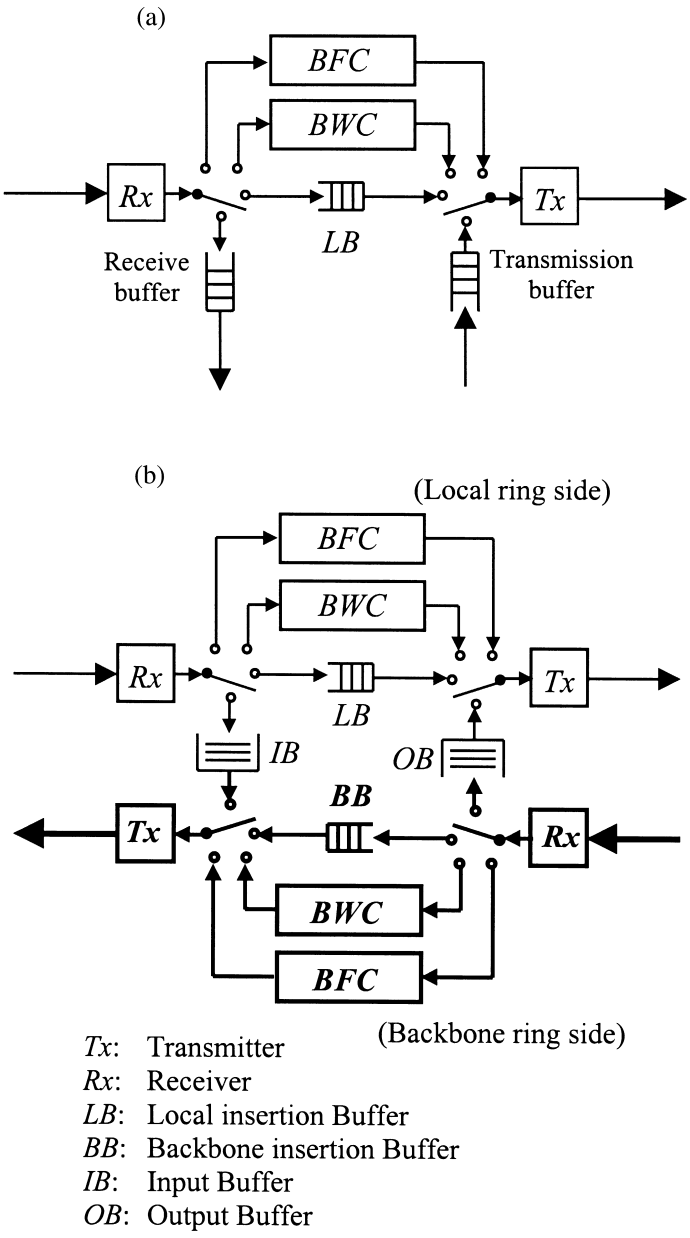


Fig. 3. The structures of (a) the user node and (b) the bridge.

the backbone ring, respectively. Additionally, it includes two BFCs and two bridge buffers for incoming and outgoing inter-traffic. Due to identical media access control mechanisms on the local and backbone rings, the two BWCs in a bridge are the same, except the BWC for backbone ring having higher processing speed. For the bridge buffers, one stores packets from local ring to backbone ring and the other stores packets in the reverse direction. The former and the latter are called the input buffer (IB) and the output buffer (OB), respectively. To prevent buffer overflow and to achieve fair usage of buffer capacity, BFC is used to control utilization of IB or OB. As BWC, BFC has two states. If input rate of 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, capacity utilization of that buffer will be stable. At this point, its corresponding BFC is doing nothing except for continuously monitoring remaining buffer capacity and staying in the passive state. Otherwise, the BFC switches to the active state to be an initiator and regulates those input traffic flows fairly by starting a series of cyclic reservation cycles.

Based on the address scheme and the bridge structure described above, 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 a higher-level single backbone ring so that a 3-level single hierarchical ring is constructed. Evidently, constructions of multiple-level hierarchy associated with dual ring variations are also possible.

3. Reservation-based control protocols

In this section, a generic method for cyclic reservation-based access control to multiple network resources is presented in advance. Subsequently, the two parts of CRMA-HR, namely BwCP and BfCP, are described. Since these two control protocols operate independently, multiple reservation cycles for different resources may occur simultaneously. Thus, coordination of these cycles is addressed. A head-of-line blocking phenomenon may happen in case several inter-traffic flows interleave from a source bridge. The resolution of the head-of-line blocking is discussed. Since there are a few variations for the configuration of hierarchical rings, the application to these variations are demonstrated in detail.

3.1. A generic resource reservation mechanism

As mentioned earlier, network bandwidth and buffer capacity are two different network resources that can facilitate communications in hierarchical ring networks. To avoid possible access conflicts, the resources allocated to network nodes can be done by way of cyclic reservation. The method is organized into cycles of three phases: reservation, scheduling, and confirmation. In each phase, command slots, data slots, counters, and variables are involved for protocol operations. The command slots include ReSerVe and ConFirM slots, abbreviated to RSV and CFM respectively. RSV is used to collect the information for requesting and reserving resources from all nodes within a component ring. 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. The data slots include FG, BG, and FR slots. Depending on the working status, a data slot can be a FG, BG, or FR slot. Being a FR slot, it is used for

allocating resources. The counters include Transmission Counter and Request Counter, abbreviated to *TC* and *RC* respectively. For *RC*, it is increased according to the number of new packets entering a 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 must be employed. The variables used include one fairness threshold (abbreviated to *FT*), Reservation_cycle, Bw_short_flag, Bf_short_flag, and *H*. The value of *FT* is computed by a fairness threshold computation algorithm in the scheduling phase and then carried by CFM for broadcasting. For Reservation_cycle, it is the same as the slot field Reservation_cycle and has the value of either **none, bandwidth** or **buffer**. Note that, in 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 field value in one slot, the expression 'field.Slot' is employed. For instance, the value of *FT* in a CFM is denoted as '*FT*.CFM'. Additionally, a few simple functions are used such as CopyToMemory, Compute_bw_FT, Compute_bf_FT, Remove, and MarkReserve. Their meanings are rather intuitive from their naming and will not be explained further. To ease the subsequent descriptions, assume that there are n nodes in a component ring (e.g., $(n-1)$ user nodes and one bridge for local rings or n bridges for backbone ring) and every packet transmitted occupies one slot exactly. In normal status, one resource used by any 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 component ring (see step I1). For other nodes, they execute the responder algorithm as the responders while the RSV is received. At this point, the information for resource request counted by the pair (TC, RC) is copied into the payload of the 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[((TC_i, RC_i)$i = 1,..n,, i \neq k$).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 (($FT - TC_k$) > 0)
FG^{-} ⊕ packet → BG^{+}; /*confirmations for initiator*/
$TCk = TC_k + 1$; $RC_k = RC_k - 1$; /*update counters*/
} **else** MarkReserve[FG^{-}] → FR^{+}; /*resource allocation*/
}
$TC_k = 0$; /*reset transmission counter*/
}
(I6) **if** (CFM^{-}) Remove[CFM];

Responder algorithm (responder *i*)

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

In the scheduling phase, node k copies the pairs (TC_i, RC_i)$i = 1, \ldots, n$, $i \neq k$ into memory and computes the value for *FT* (steps I2 and I3) while the RSV circulating back. The *FT* value is computed by invoking the bandwidth or buffer *FT* computation algorithm. As the scheduling is in progress, the network can be completely utilized by allowing the nodes to access FG slots.

Nevertheless, this cannot be applied 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 to broadcast the *FT* value (Step I4). Following the CFM, it marks *H* FG slots as FR slots (Step I5). At the same time, the first $(FT - TC_k)$ FR slots are used to convey the packets pending in the transmission buffer of the initiator. For the 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 gets the confirmation to access FR slots or the deferment to throttle the access of succeeding FG slots (step R3). That is, if $FT > TC_i$, the confirmation of $(FT - TC_i)$ FR slots can be used to transmit the packets pending in the transmission buffer. Otherwise, to keep the resources shared fairly, the access opportunity to the succeeding $(TC_i - FT)$ FG slots must be left to the downstream nodes. Particularly, such a fair access control benefits the resolution of network congestion [6]. Finally, the initiator removes the CFM from the network (step I6). If the situation of resource shortage were continued, the initiator would start a new cycle again.

3.2. Bandwidth control protocol

Roughly speaking, the BwCP protocol consists of the initiator, responder, and bandwidth *FT* computation algorithms. Here, reservation cycle is called bandwidth cycle. As described earlier, every node always monitors utilization of network bandwidth. While a node detecting poor access to unavailable network links, its *BWC* switches to the active state at once and executes the initiator algorithm with Reservation_cycle set to **bandwidth**. When a RSV circulates, *BWC* of other nodes receiving the RSV will switch to the active state as well and execute the responder algorithm. If more than one node declares itself as the initiator at a time, the arbitration must be taken. Although a number of arbitrating strategies might exist, the simplest way is to compare their node addresses. That is, if an initiator receives another RSV with larger source node address, then it must change itself as the responder at once and drop the issued RSV. By this way, it guarantees only one initiator existing. The other nodes are the responders. BwCP can be applied to backbone or local rings without differences.

As regards the computation of *FT* value, the algorithm described below is based on the principle of computing the minimum summation of either confirmations or deferments over all nodes in a component ring. Since it causes a minimal degree of control intervention to 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, ..., TC_n by iterations. As 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

$Tmax = \max\{TC_1 TC_2, \ldots, TC_n\};$
$Tmin = \min\{TC_1, TC_2 \ldots, TC_n\};$
sum = **infinity;**
for $(i = Tmin, i \leq Tmax, i = i + 1)$ { /*compute the optimal threshold (*FT*)*/

```
    temp = ∑_{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 ≤ 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 in a bandwidth cycle, we take the example scenario given in Fig. 4. 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 to collect the parameter pairs (TC_i, RC_i), where $i = 2$, 3, or 4. For convenience, (TC_4, RC_4) represents 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 $Tmin = 2$ to $Tmax = 7$, FT is 3 and then the number of slots reserved (H) is 1. At this point, $U_{1,1}$ issues a

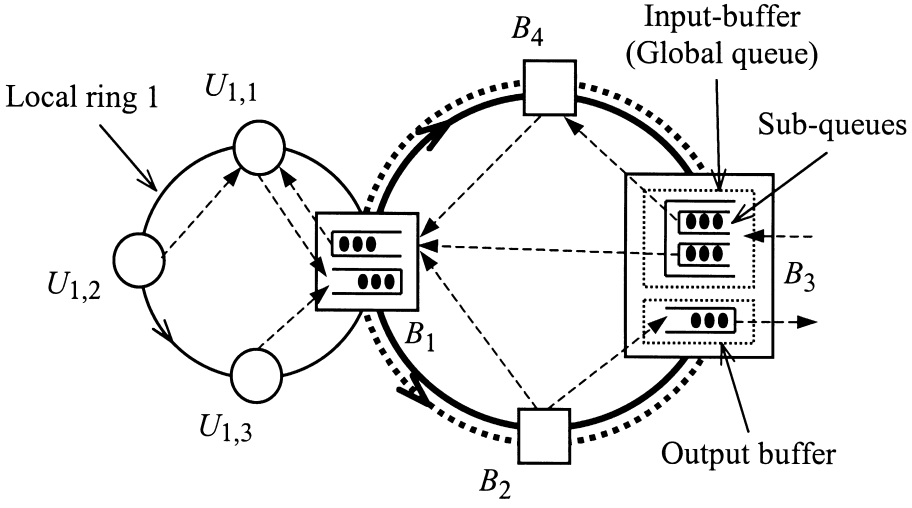
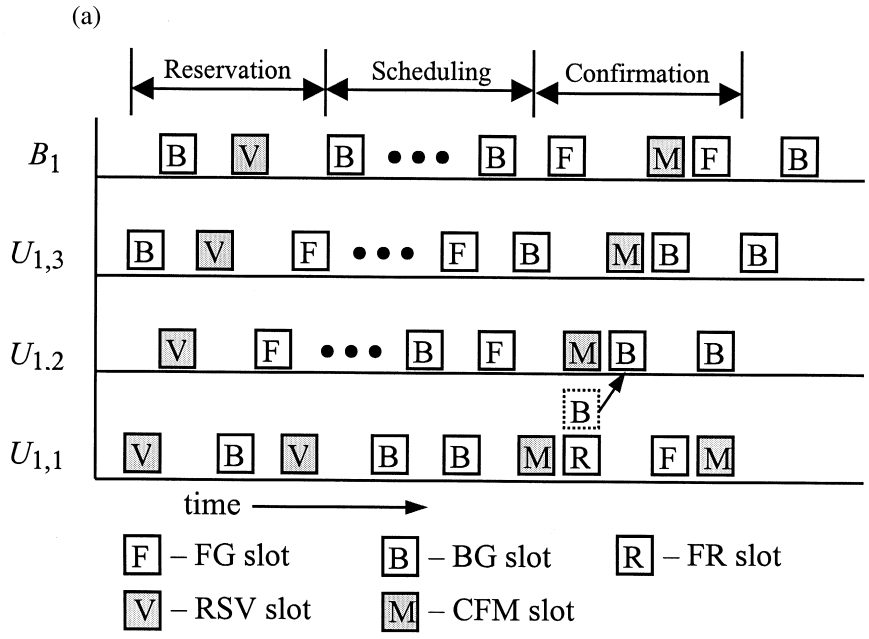


Fig. 4. An operational scenario for the bandwidth and buffer reservation mechanisms.

CFM with $FT = 3$ and marks one FG slot as FR slot (Fig. 5(a)). However, the FR slot is used immediately by $U_{1,1}$ itself and therefore marked as BG slot.

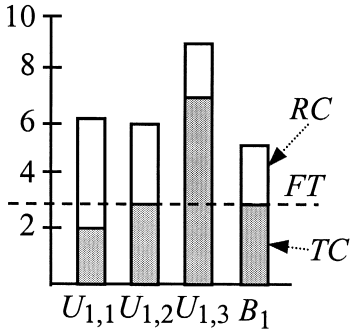
3.3. Buffer control protocol

In hierarchical networks, buffer overflow, which occurs in bridges due to heavy load or overload condition, is a crisis provided that no proper congestion control is applied. The BfCP



(b)

Iteration	1	2	3	4	5	6
Threshold	2	3	4	5	6	7
Minimum of sum	7	5	7	9	10	10



(TC_1=2, RC_1=5, TC_2=3, RC_2=3, TC_3=7, RC_3=2, TC_4=3, RC_4=2, $Tmin$=2, $Tmax$=7)

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

protocol can regulate inter-traffic through buffer (e.g., *IB* or *OB*). Basically, it consists of the initiator, responder, and buffer *FT* computation algorithms. The reservation cycle here is called buffer cycle. As stated above, the *BFC* for *IB* or *OB* in every bridge always monitors utilization of the buffer capacity counted in number of slots. Suppose that, in backbone ring, the *BFC* for *OB* of a bridge (say bridge k) detects available buffer capacity down to a level, causing traffic congestion, it switches to the active state at once and executes the initiator algorithm by setting Reservation_cycle to **buffer**. As a RSV circulating, the *BFC* of bridge i, where $i \neq k$, receiving the RSV must switch to the active state and execute the responder algorithm. At this point, it copies (TC_i^k, RC_i^k), used to trace the inter-traffic flow from bridge i to the initiator, into payload of the RSV and issues the RSV again to its downstream neighbors. Note that, in backbone ring, multiple buffer cycles initiated by different *BFC*s can happen simultaneously without mutual interference. In particular, the buffer cycles will not exclude the bandwidth cycle occurred in the same ring as well. Nevertheless, one situation is important in local rings. Since user nodes have no bridge buffers like *IB* or *OB*, the only buffer that is related to a user node is the *IB* in the bridge of the corresponding local ring (e.g., local ring 1 and B_1 in Fig. 4). Therefore, *BFC* of a user node is only necessary to execute the responder algorithm (being a permanent responder) while receiving the RSV from the *BFC* for the *IB* in the bridge under heavy load condition. Conversely, the *BFC* for the *IB* plays the role of permanent initiator. In addition, for those inter-traffic flows destined to a buffer with insufficient capacity in the scheduling phase of buffer cycles, the responder algorithm always suspends the access to the passing FG slots. This aids the avoidance of buffer overflow.

Undoubtedly, buffer capacity is finite. Without congestion control, greedy users may generate a lot of inter-traffic to occupy most capacity. At this point, the network is always congested due to insufficient buffer capacity. To keep away from network congestion, balancing buffer sharing among different transmission requests is necessary. To equalize access opportunities to a buffer, a fairness method called the less buffer's occupation first (LBOF) discipline is adopted for the buffer *FT* computation. This discipline is very intuitive. It prioritizes the access privilege for those source bridges starving to utilize the buffer capacity. To compute values of *FT* and *H*, we take *OB* as an example to describe the algorithm below. As for *IB*, the algorithm can be applied as well. First of all, the following assumptions are made. The *OB* of all bridges has the size of C slots. The capacity of the *OB* of B_k is insufficient so that the buffer cycle is started. A_k is the occupied capacity while the RSV is 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. Obviously, the *FT* computation algorithm for buffer is quite different from that for bandwidth.

Buffer *FT* computation algorithm

$m = 0$; /*m counts the number of bridges removed*/
$S = \{B_1, B_2, \ldots, B_{k-1}, B_{k+1}, \ldots, B_n\}$; /*the set of all bridges except B_k^*/
while (**true**) {

$$Topt = \left\lfloor \frac{(C - A_k) + \sum_{\forall B_i \in S} TC_i^k}{n - 1 - m} \right\rfloor \quad ; \ /^*\text{compute the threshold}\ ^*/$$

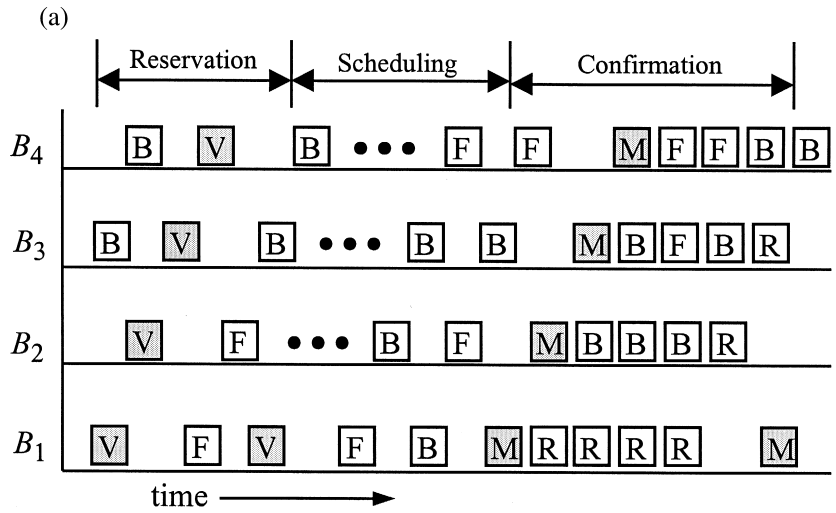
```
   EndOfLoop = true; /*a test flag for ending the while loop */
     for (i = 1, (i≤n) ∧ (i≠k), i = i + 1) {
       if((Topt − TC_i^k)≤0) ∧ Bi ∈ S) {
         S = S − {B_i}; /*this is the set operation to remove B_i*/
         m = m + 1; /*count one removed bridge*/
         EndOfLoop = false;
       }
     }
   if(EndOfLoop = true)
     break; /*leave the while loop*/
   }
   FT = Topt;H = 0;
   for (i = 1, (i≤n) ∧ (i≠k), i = i + 1) {/*count the reserved slots*/
     if (Topt > TC_i^k) {
       if ((Topt − TC_i^k) > RC_i^k)
         H = H + RC_i^k;
       else H = H + (Topt − TC_i^k);
     }
   }
   return(FT, H);
```

In the beginning, 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., $(Topt - TC_i^k) \leq 0$. Then, those bridges are removed from *S*. Subsequently, *Topt* is computed again by considering the remaining bridges in *S* and the remaining size of destination *OB*, viz. $(C - A_k)$ slots. The iteration is repeated until it stops under the condition $(Topt - TC_i^k) > 0$ for all remaining nodes in *S*. The *FT* value is determined. In this case, avoidance of buffer overflow and fair usage of buffer capacity can both be achieved. In addition, the total number (H) of slots reserved is also computed at the same time. The values of *FT* and *H* are then sent back to the initiator algorithm.

To illustrate the determination of *FT* and *H* using the algorithm in a buffer cycle, we take the example scenario given in Fig. 4. Assume that B_2, B_3 and B_4 have inter-traffic flows to the *OB* of B_1 and 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 values of the 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. Thus, *FT* is 6 and then 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.

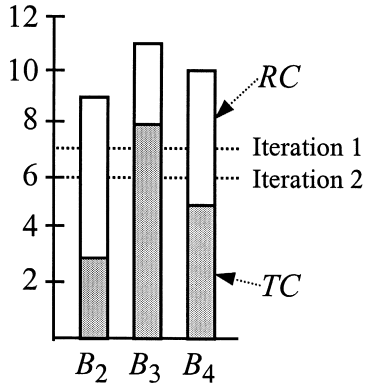
3.4. Coordination of multiple reservation cycles

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



(b)

	Iteration 1			Iteration 2		
	B_2	B_3	B_4	B_2	B_3	B_4
Confirmation	4	-1	2	3	–	1
Topt	7			6		
$S =$	$\{B_2, B_3, B_4\}$			$\{B_2, B_4\}$		



$$(TC_2{}^1=3, RC_2{}^1=6, TC_3{}^1=8, RC_3{}^1=3, TC_4{}^1=5, RC_4{}^1=5, C=10, A_1=5)$$

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

case, if $U_{1,3}$ determines the deferments for the buffer cycle, then it has to ignore the subsequent activity in the bandwidth cycle and work in the buffer cycle in order to avoid possible buffer overflow. For the second case, if $U_{1,3}$ determines the confirmations for both of the buffer and bandwidth cycles, then the smaller quantity of confirmation is preferred so as to maintain fair access of bandwidth as well as avoid 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 fair access of bandwidth. The same coordination approach can be applied as well to the bandwidth and buffer cycles simultaneously occurring in the backbone ring.

3.5. Cancellation of head-of-line blocking

In general, *IB* of a bridge (e.g., the *IB* of bridge B_3 in Fig. 4) may store packets belonging to several inter-traffic flows destined for different destination bridges. If the *OB* of one of the destination bridges is getting heavily used, the inter-traffic flow destined for that bridge would block all succeeding inter-traffic flows in the *IB* until the marked FR slots arrive. More seriously, if the capacity of *OB* of two or more destination bridges were insufficient, a head-of-line blocking phenomenon would happen due to waiting for different marked FR slot sequences. 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 arrives. Again, suppose that both cycles for the *OB*s of B_1 and B_3 are started simultaneously. The problem happens in the *IB* of B_2. Actually, frequent occurrence of the head-of-line blocking is much harmful to the performance of global network.

To tackle this issue, a multiple sub-queue scheme with a multiple linked list structure can be adopted. This scheme constructs a multiple linked list where one linked list is used to hold a sub-queue of packets destined for one specific bridge. Simultaneously, the global FIFO order for the arriving packets is also maintained. When FG slots pass through, a packet is selected for transmission from those sub-queues in the global FIFO order. However, for the FR slots of one specific reservation cycle passing, the corresponding sub-queue is selected and the data packet is transmitted. For example, in Fig. 4, two sub-queues storing the inter-traffic flows to B_1 and B_4 in the *IB* of B_3 are drawn. 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 sub-queue scheme, we skip the detailed explanations.

3.6. Application to dual ring configurations

A local or backbone ring can be a dual ring. Mixing with the structure of single ring, several variations for hierarchical rings may exist. Generally, they include the configurations of single backbone ring connecting dual local rings, dual backbone ring connecting single local rings, and all component rings being dual rings. Clearly, the two single rings of a dual ring may rotate in either the same or in opposite directions. To have shorter transmission path for a packet to its destination, assume that the two single rings of every dual component ring in a

hierarchical ring rotate mutually in opposite directions. From the practical viewpoint, the configuration of single backbone ring connecting dual local rings is unrealistic because a dual local ring can bear much heavier traffic than a single local ring, making the single backbone ring in heavy congestion.

For the configuration of all component rings being dual rings, it is the dual hierarchical ring configuration, or the node-to-node combination of two equivalent single hierarchical rings. Due to the dual ring property, there are multiple transmission paths that can be chosen for a packet to its destination. For instance, for a packet belonging to intra-traffic, it has two paths that can be selected by a user node. On the other hand, a packet belonging to inter-traffic will pass through source local ring, backbone ring, and destination local ring. Hence, there are up to eight possible paths that can be chosen. In general, for an m-level dual hierarchical ring, the longest path that a packet passes through is along $(m-1)$ level hierarchy upward, innermost backbone ring, and $(m-1)$ level hierarchy downward. As a result, there are up to $2^{(2*m-1)}$ possible paths. Regardless of the number of paths, the routing strategies used by a node may rely on the principle of shortest path or the condition of traffic load. Anyway, the binary selection based on the dual ring structure for the routing path of a packet is straightforward and simple in a user or bridge node.

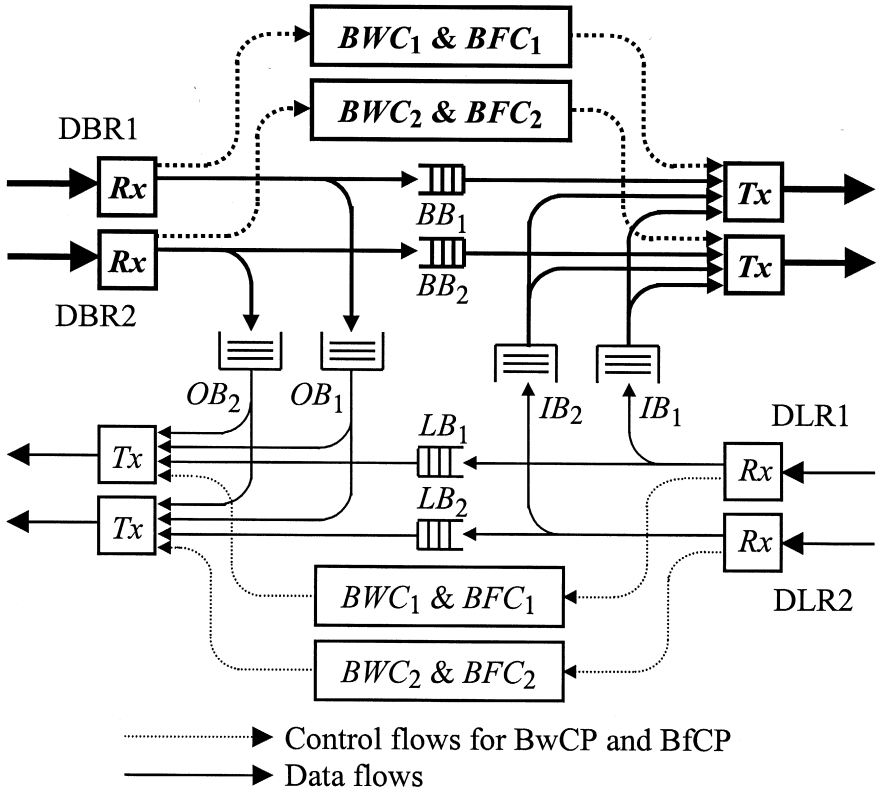


Fig. 7. Logical traffic flows in a bridge node.

In a dual hierarchical ring, the BwCP and BfCP protocols can be applied directly without any further enhancement. Since a bridge node is used to connect two backbone rings and two local rings together, there are double sets of *BWC*s, *BFC*s, *IB*, and *OB* embedded in the structure of bridge nodes. In Fig. 7, the logical traffic flows, including the data flows for normal traffic and the control flows for BwCP and BfCP, are illustrated. As with in a single hierarchical ring, a control flow is formed by a series of command slots. Note that as the previous assumption, the first ring DBR1 and second ring DBR2 of dual backbone ring (or the first ring DLR1 and second ring DLR2 of dual local ring) are rotated in opposite directions though they are sketched logically in the same way in Fig. 7. For the side of backbone ring, the data flows in DBR1 (or DBR2) destined for local ring are forwarded to BB_1 (or BB_2). Otherwise, they will be forwarded to OB_1 (or OB_2) for transmission to the next bridge node. The data flows kept temporarily in OB_1 (or OB_2) are routed to either DLR1 or DLR2 depending on the routing strategies described earlier. If the data flow in OB_1 or OB_2 is routed to DLR1 (or DLR2), it must be multiplexed with the data flows kept in LB_1 (or LB_2). For the side of local ring, the same situation happens that the data flows from DLR1 (or DLR2) are kept in IB_1 (or IB_1) and then can be routed to either DBR1 or DBR2. Otherwise, they will be forwarded to LB_1 (or LB_2). It is worthy to note that the different control flows never cross the ring boundary as sketched in Fig. 7. For example, the control flow of BwCP for DBR1 (or DBR2) and the control flow of BfCP for OB_1 (or OB_2) travel along DBR1 (or DBR2) independently. As described above, the same situation happens for the control flow of BwCP for DLR1 (or DLR2) and the control flow of BfCP for IB_1 (or IB_2). Since the operations of BwCP for each single ring component and the operations of BfCP for each bridge buffer work independently, the traffic congestion due to the resources in shortage can be resolved by the mechanisms of BwCP and BfCP.

In case of the configuration of dual backbone ring connecting single local rings, a bridge node is used to connect the dual backbone ring and one local ring together. Obviously, a packet belonging to intra-traffic has one transmission path to its destination. While belonging to inter-traffic, it has only two transmission paths to select. Therefore, the routing strategy for this configuration should be simpler and more straightforward than for the dual hierarchical ring described above. In this configuration, the BwCP and BfCP protocols can be applied directly as well. As with in a dual hierarchical ring, a bridge node contains two *OB*s buffering data flows from the dual backbone ring to a local ring and one *IB* buffering data flows from the reverse direction. To conduct numerous access requests to the resources, there are two *BWC*s for dual backbone ring, one *BWC* for single local ring, two *BFC*s for *OB*s and one *BFC* for *IB* in a bridge. Because of the independent reservation cycles of these *BWC*s and *BFC*s, different control flows never cross the ring boundary as with the situation of dual hierarchical ring drawn in Fig. 7. For example, the control flow of BwCP for dual backbone ring (or single local ring) and the control flow of BfCP for *OB*s (or *IB*) travel along backbone rings (or local ring) independently. In any case, the traffic congestion due to the resources in shortage somewhere in one network based on this configuration could be resolved according to BwCP and BfCP. Since the structure of dual backbone ring can bear more inter-traffic, this configuration potentially allows more single local rings to connect.

4. Performance evaluation

To analyze the performance characteristics of CRMA-HR, a simulation model is described below. Because of the hierarchical ring configuration, the ratio of inter-traffic to intra-traffic would affect the performance results a lot. To evaluate such a large-scale and ultra high-speed MAN, delay and throughput versus network length and the number of user node based on various traffic conditions will be of particular interest.

4.1. Simulation model

Our simulation model is composed of four parts: the network model, the user node model, the bridge model, and the load generation model. The network model is based on a single hierarchical ring with all nodes placed equidistantly. In accordance with a standardized transmission line, the transmission rate of local rings is set to 1.2 Gbit/s (STM-8) and the rate of backbone ring is set to 2.5 Gbit/s (STM-16). Transmission medium is assumed to be the optical fiber with 5 μs/km propagation delay. The node latency, which occurs due to storing and checking slot header, associated with either bridges or user nodes is set to 1 slot (72 octets). The error rate of transmission medium is not considered. For bridges and user nodes, the node structures have been illustrated in Fig. 3. Each *BWC* or *BFC* is simplified to be as a command delay buffer and a set of registers used to count transmitted slots, slot requests, confirmations, and deferments. In fact, each user node is modeled by a set of transmitter, receiver, insertion buffer, transmission buffer, *BWC*, and *BFC*, and each bridge by two sets of transmitter, receiver, insertion buffer, *BWC*, *BFC*, input buffer, and output buffer.

The following load models are used in the 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 silence period is determined by an exponential distribution. Regarding the offered load, the length of silence can be varied to control traffic density in this model. The heavy load model is used in the simulations with respect to network size and the number of user nodes supported. This model is implemented by always having a new packet to send in transmission buffer. Another important factor that can affect the simulations is traffic distributions. Supposing that each group of user nodes within a local ring is treated as a separate communication community, the traffic distribution resulting from different 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 a ratio for the traffic distribution with locality. We will investigate the performance of single hierarchical ring according to different locality ratios.

To realize the performance of CRMA-HR, network throughput, access delay, and averaged total delay are calculated in the simulations. The network throughput is defined as the total number of packets received by all user nodes divided by the number of packets that can be transmitted by the link of the network within a second. The access delay is defined as the time between the arrival of a packet at the head of transmission buffer until the first bit is transmitted. The transfer delay of a packet is defined as the time between the transmission of

first bit from source node until the last bit is received by destination node. The total delay of a packet is defined as the sum of the access and transfer delays. Hence, the averaged total delay is given by averaging over all packets generated during the simulation time. The parameters used in the simulations are summarized in Table 1. As for the total number of slots reserved (H) for each cycle of either bandwidth or buffer, it can be determined dynamically by the bandwidth or buffer FT computation algorithms in the simulations. 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.

4.2. Performance results

In a single hierarchical ring, the user nodes generate both intra- and inter-traffic. For intra-traffic, all nodes are assumed to send to all destinations uniformly, except to themselves, in a local ring. For inter-traffic, it has the same assumption about uniform distribution to whole user nodes within a single hierarchical ring, except to those nodes in the same local ring with the source. To investigate the performance under different locality properties, the ratios 50, 60, 70, 80, and 90% are chosen. Also, the network throughputs of local, backbone, and global rings are normalized with respect to the link capacity of local rings, i.e. 1.2 Gbit/s, for the purpose of comparison.

To the first simulation, the network configuration consists of ten local rings (with five user nodes and one bridge in each local ring) and a 70-km network with all nodes placed equidistantly. The input load is based on the burst-silence model. Fig. 8 illustrates the growth of the averaged total delay versus the global throughput according to five locality ratios. The asymptote of lower locality ratios shows lower throughput and vice versa. To explain this

Table 1
Simulation parameters

Parameters	Values
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
Node latency	1 slot
Propagation delay	5 μs/km
Slot length	72 octets
Scheduling latency (bandwidth)	local rings (10 μs), Backbone ring (5 μs)
Scheduling latency (buffer)	Input buffer/output buffer (5 μs)
Load models	Burst-silence/heavy

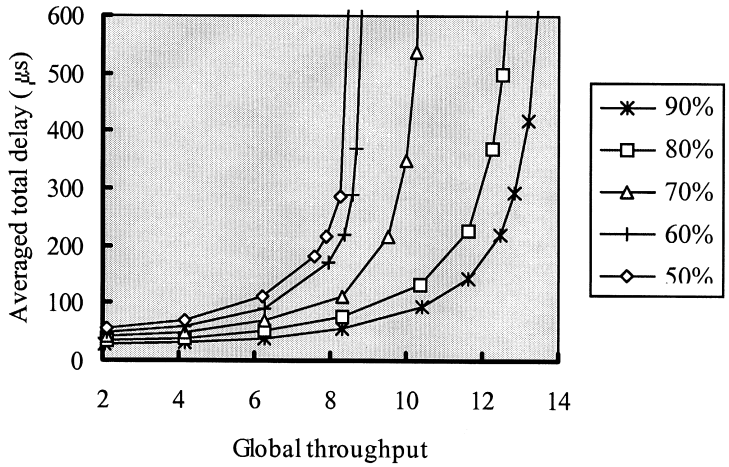


Fig. 8. Averaged total delay vs. global throughput.

situation, consider two extreme cases: zero and hundred percent locality ratios. For the zero percent, there is no intra-traffic by definition. All traffic transferred to their destinations need to be switched through the backbone ring. Thus, the backbone ring would be saturated very quickly by lots of inter-traffic. At this point, the global throughput is lowest. For the hundred percent, all traffic belongs to intra-traffic, transferring to their destinations inside the respective local ring. The global throughput becomes the summation of the network throughput of all local rings. As a result, higher locality ratio benefits the global throughput of hierarchical ring networks. From another point of view, the growth of the global throughput is dominated by the bandwidth of the backbone ring in lower locality ratios (e.g., 50–70%). If the bandwidth of

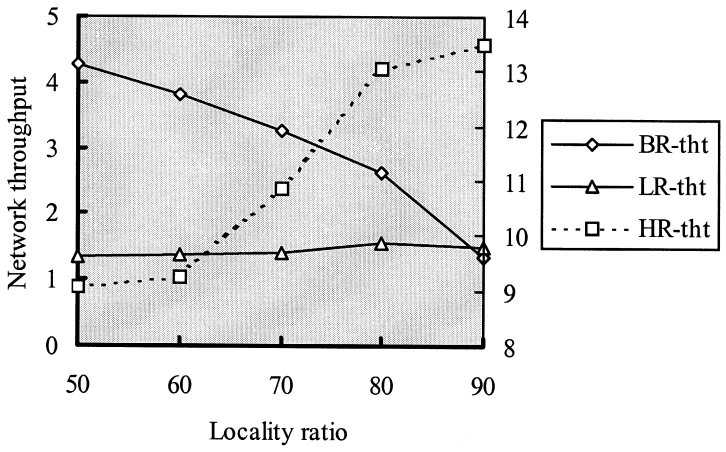


Fig. 9. Network throughput vs. locality ratio (BR-tht: backbone ring's throughput; LR-tht: local ring's throughput; HR-tht: hierarchical ring's throughput).

the backbone ring is enlarged, more inter-traffic can be switched for increasing the global throughput.

For the following simulations, the input load is based on the heavy load model. In Fig. 9, the curves of the simulated throughputs for backbone, local, and global rings are depicted respectively, according to the locality ratios, where the throughput of local ring is an average over all local rings. Owing to the different scales, the throughput of backbone and local rings are sketched with respect to the left vertical axis and the global throughput to the right vertical axis. As depicted, the throughput of local ring varies little (within the range of 1.33–1.56), but for the backbone ring, its throughput decreases a lot due to very less inter-traffic. It is interesting to see that the value of the averaged throughput of local rings multiply the number of local rings and minus the throughput value for the backbone ring is very approximate to the value of the global throughput drawn in Fig. 9. Precisely, the situation is twofold: the throughput of the backbone ring is produced by the inter-traffic from all local rings through bridges and normally the throughput of a local ring includes the part of inter-traffic received by the bridge of the local ring. In that case, the global throughput can be approximated by the net result of the multiplication by the throughput value of local rings and the number of local rings minus the value of duplicate throughput occurred in the backbone ring.

To cover a wider communication area, the MAC protocol of a network must be independent of its network size. This item is related to the well-known *a-parameter* [2], which is defined as the ratio of the propagation delay and the packet transmission time itself. As a rule, network throughput degrades with increasing this ratio. Nevertheless, this ratio always grows with higher bit rate or larger network size. Therefore, any protocol that is sensitive to this parameter limits its usefulness to networks which are only up to a certain size. To evaluate the sensitivity of CRMA-HR to network size, the relationships of the global throughput and the access delay with respect to the network size for different locality ratios are simulated and drawn in Fig. 10. In this simulation, the simulation conditions are the same as above except for the network length between any two nodes varying from 1–20 km (i.e., the network size from 70–1400 km). As can be seen, the global throughput for each locality ratio shrinks slightly with respect to the growth of the network size in Fig. 10(a). Especially, the global throughput for 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 the case of low locality ratio. As for the averaged access delay with respect to the network length, the delay for each locality ratio varies slightly as well in Fig. 10(b). Notably, the delay grows as the increase of locality ratio except for 90%. There is a conversion phenomenon between 80% and 90%. To see this, the throughput of local ring for 80% is higher than the throughput for 90% as depicted in Fig. 9. Because the traffic for 80% is denser than for 90% in a local ring, the denser traffic makes the access delay of 80% longer.

For a large network with lots of user nodes, the MAC protocol of such a network must be highly efficient. Otherwise, when the network scales up to a certain size, it is difficult to cope with plenty of user applications. For a hierarchical ring, the variation of the node number in component rings may affect the network performance a lot. To investigate the performance of CRMA-HR versus the number of user nodes, two network configurations are considered. In configuration 1, ten bridges are used in the backbone ring, and the number of user nodes in each local ring varies from 5 to 25. On the contrary, in configuration 2, ten user nodes are

used in each local ring, and the number of bridges varies from 5 to 25. Whatever the configuration is, the total number of user nodes varies from 50 to 250. Alternatively, the locality ratios 60% and 80% are chosen in the demonstration of performance impact. Fig. 11 shows the relationships of the global throughput and the access delay with respect to the number of nodes. For the combinations of two configurations and two locality ratios, four curves (denoted as C1, C2, C3 and C4) are sketched in Fig. 11(a) and (b), respectively.

In Fig. 11(a), two types of comparisons are made from the configurations and the locality ratios. For the configurations, the global throughput for configuration 2 (curves C3 and C4) is better than for configuration 1 (curves C1 and C2) as the number of node exceeds one hundred. Note that since the number of local ring varies increasingly in configuration 2, it is beneficial for the growth of the global throughput until the backbone bandwidth is unavailable. As for configuration 1, the number of local ring is fixed so that the growth of the

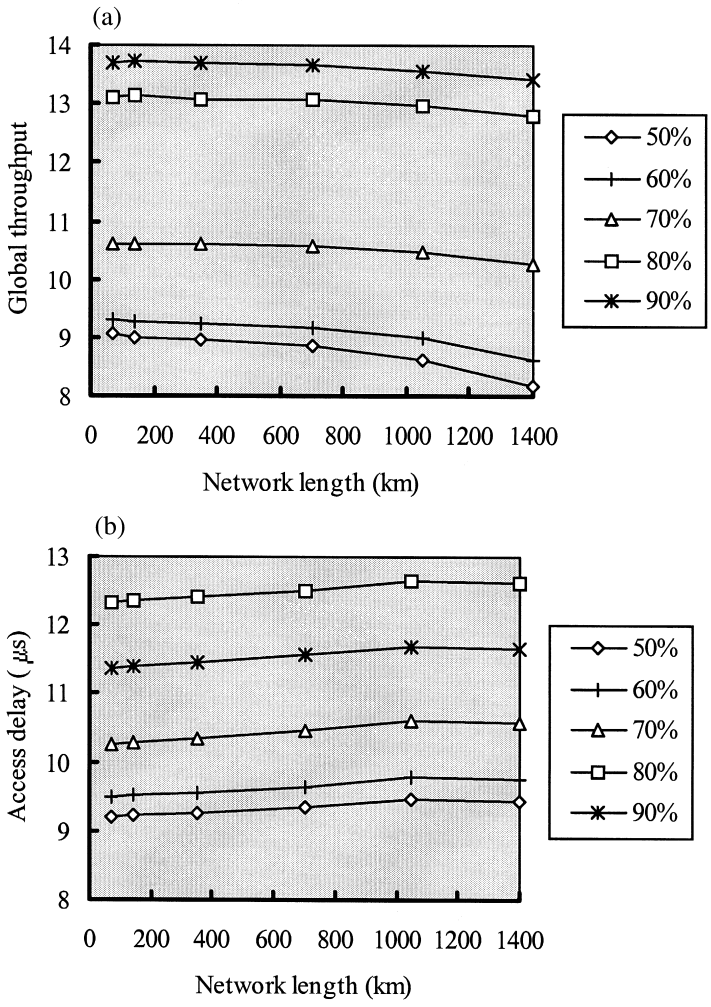


Fig. 10. (a) Global throughput vs. network length and (b) access delay vs. network length.

global throughput is bounded by a limitation. Whichever the backbone or local rings may saturate under heavy traffic condition, the throughput limitation will be approached rapidly. For the locality ratios, it is obvious that curves C2 and C4 (80%) show superior to C1 and C3 (60%). The above simulations are because the backbone ring is saturated by the inter-traffic for 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 node. Next, the saturation of the backbone ring starts from one hundred nodes for C3 and one hundred and fifty nodes for C4. In Fig. 11(b), the comparison is as above according to the configurations and the locality ratios. Explicitly, 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

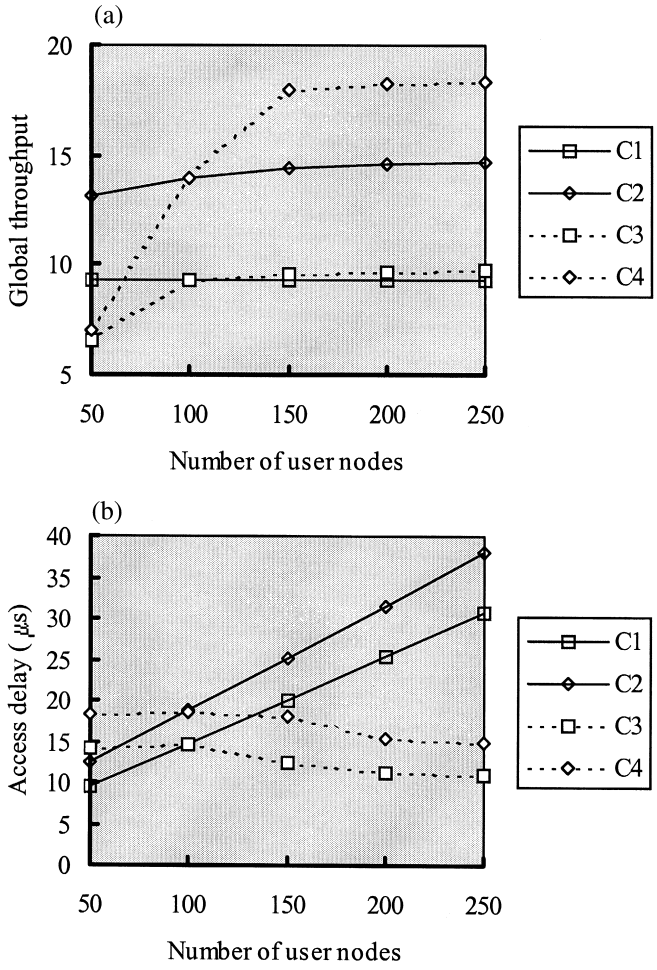


Fig. 11. (a) Global throughput vs. number of user node and (b) access delay vs. number of user node (C1: Configuration 1 with locality ratio 60%; C2: Configuration 1 with locality ratio 80%; C3: Configuration 2 with locality ratio 60%; C4: Configuration 2 with locality ratio 80%).

viewpoints of access opportunity and the number of nodes. In configuration 1, the relative access opportunity of each node to a local ring decreases by degrees with respect to the growing node number. Therefore, the access delays of C1 and C2 grow accordingly in a linear way. Alternatively, since the node number of local rings is fixed in configuration 2, the access opportunity of user node varies slightly. At the same time, increasing the number of local ring, the traffic density, or equivalently the shared traffic load, in each local ring becomes less so that the access delay can be shorter. Hence, it makes the access delays of C3 and C4 change in a slightly decreasing way according to the node number. As for the effect of the locality ratio to the access delay, it is shown that higher ratio has longer delay due to dense intra-traffic and few access opportunities in local rings.

5. Conclusions

In this investigation, we have presented a resource reservation approach to design the MAC protocol, CRMA-HR, for a large-scale and ultra high-speed MAN based on the hierarchical ring configuration. The network can be used to support a great number of users with highly aggregated bandwidth and covers larger geographical area than traditional MANs. For CRMA-HR, it consists of BwCP and BfCP. The BwCP protocol controls the utilization of network bandwidth of component rings, including the backbone ring, under heavy or unbalanced workload. While incorporating with slotted and buffer insertion access mechanisms, high network utilization, enforced fairness, and bounded access delay can be achieved. For BfCP, it controls the utilization of bridge buffers (viz., *IB* and *OB*). Without this protocol, the network congestion due to buffer overflow can block inter-traffic flows so that the global network utilization decreases, and the averaged transfer delay becomes unpredictable. It is worthy to note that these control protocols adopt the same resource reservation scheme, which can simplify the development of network interface units and reduce the implementation cost. In addition, it is inevitable that CRMA-HR has to face with different network variations, e.g. double hierarchical rings, multiple level ring networks, and so forth. For double hierarchical rings, CRMA-HR has been shown to work exactly and very well. Particularly, an inherent advantage of hierarchical ring configurations is its self-similarity for extending to the multiple level ring networks. Whatever the number of level may be, the network resources used to facilitate communications are still network bandwidth and buffer capacity. Due to the flexible addressing scheme, CRMA-HR can easily be extended to a multiple level hierarchical ring environment.

To investigate the performance of CRMA-HR, several simulations were designed, implemented, and performed. From the simulation results, it shows the optimistic global throughput and averaged transfer delay with respect to the locality traffic distribution. Especially, its related performance characteristics can fit to network scalability in terms of the network size and the number of user nodes.

In summary, our approach for a large-scale and ultra high-speed MAN has the properties of simple routing, straightforward media access, high connectivity for attachments, and self-similarity of network topology for extension. Furthermore, CRMA-HR can satisfy the fairness

requirement and operate independent of the network size and the number of user nodes. To reduce the complexity of MAC protocol, a resource reservation mechanism is developed especially to facilitate the access control for different network resources. Finally, some successive work is unfinished. For instance, the optimization of arrangement for multiple reservation cycles and other approach for resolving and optimizing the head-of-line blocking problem still need to be pursued.

References

[1] Ajmone Marsan M, Albertengo G, Casetti C, Neri F, Panizzardi G. On the performance of topologies and access protocols for high-speed LANs and MANs. Comput Networks & ISDN Syst 1994;26:873–93.
[2] van As HR. Media access techniques: the evolution towards terabit/s LANs and MANs. Comput Networks & ISDN Syst 1994;26:603–56.
[3] van As, HR, Lemppenau WW, Schindler HR, Zafiropulo P. CRMA-II: a MAC protocol for ring-based Gb/s LANs and MANs. Comput Networks & ISDN Syst 1994;26:831–40.
[4] van As, HR, Lemppenau WW, Zafiropulo P. Performance of CRMA-II: A reservation-based fair media access protocol for Gbit/s LANs and MANs with buffer insertion. In: Proc. EFOC/LAN'92, Paris, France, Jun. 1992. p. 162–9.
[5] Brassil J, Choudhury AK, Maxemchuk NF. The Manhattan Street Network: a high performance, highly reliable metropolitan area network. Comput Networks & ISDN Syst 1994;26:841–58.
[6] Douligeris C, Kumar LN. Fairness issues in the networking environment. Comput Comm 1995;18(4):288–99.
[7] Gerla M, Kleinrock L. Congestion control in interconnected LANs. IEEE Network 1988;2(1):72–6.
[8] Imai K, Ito T, Kasahara H, Morita N. ATMR: asynchronous transfer mode ring protocol. Comput Networks & ISDN Syst 1994;26:785–98.
[9] Lee WT, Kung LY. The optimized architecture of hierarchical ring networks. In: Proc. IEEE SICON'97, Singapore, Apr. 14–17. 1997. p. 247–59.
[10] Lemppenau WW, van As, HR, Schindler HR. A 2.4 Gbit/s ATM implementation of the CRMA-II dual-ring LAN and MAN. In: Proc. EFOC/LAN'93, The Hague, The Netherlands, Jun. 1993. p. 274–81.
[11] Maxemchuk NF. The Manhattan Street Network. In: Proc. IEEE GLOBCOM'85, Dec. 2-5. 1985. p. 255–61.
[12] Maxemchuk NF. A comparison of linear and mesh topologies — DQDB and the Manhattan Street Network. IEEE J on Selected Areas in Comm 1993;11(8):1278–89.
[13] Maxemchuk NF. 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, Jun. 1990. p. 209–33.
[14] Meuser T. Performance comparison of media access protocols for Gbit/s networks in the local area. Comput Comm 1995;18(1):4–14.
[15] Nassehi MM. CRMA: an access scheme for high speed LANs and MANs. In: Proc. *IEEE ICC'90*, Atlanta, GA, 16-19 Apr. 1990. p. 1697–702.
[16] Ofek Y. Overview of the MetaRing architecture. Comput Networks & ISDN Syst 1994;26:817–29.
[17] Ross FE. An overview of FDDI: fiber distributed data interface. IEEE J Selected Area Comm 1989;7:1043–51.
[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. He has been a researcher of Telecom Lab., Chunghwa Telecomm Co., Ltd since 1989. His research interests include performance evaluation, communication protocol design, high-speed networks, and protocol engineering.

Jun-Yao Wang received the B.S. degree in Computer Science and Information Engineering from Tatung University in 1992. In 1994, he received the M.S. degree in Electrical Engineering from National Cheng-Kung University, Taiwan. He is also a Ph.D. candidate. His current interests include high-speed multi-access protocol, local area networks, and performance evaluation.

Wen-Shyang Hwang received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from National Cheng-Kung University, Taiwan, in 1984, 1990 and 1996, respectively. He is an associate professor of Electrical Engineering, National Kaohsiung Institute of Technology, Taiwan. His current research interests are in multi-channel WDM networks, performance evaluation, QoS, RSVP, WWW database applications.