A PREEMPTIVE PRIORITY SCHEME FOR IEEE 802.14 HFC NETWORKS

Jenhui Chen
Department of Computer Science and
Information Engineering, Chang Gung University
Tao-Yuan, Taiwan, R.O.C.
email: jhchen@csie.cgu.edu.tw

Shiann-Tsong Sheu and Sheng-Kun Shen
Department of Electrical Engineering
Tamkang University
Tamsui, Taipei, Taiwan, R.O.C.
email: stsheu@ee.tku.edu.tw

ABSTRACT
The hybrid fiber coax (HFC) technology enables the conventional cable-television (CATV) network to provide subscribers with Internet access services. In this paper, we propose a new *preemptive priority scheme* (PPS) for IEEE 802.14 hybrid fiber coax (HFC) networks with the *intelligent nodes* (INs). The INs are placed between the headend controller and stations. By using INs, that stand for downstream subscribers to contend for the demand resources, the collision probability and the collision resolving period can be reduced [4]. In this paper, we further extend such network architecture to support multi-priority access. In each IN or individual station, the proposed PPS will prevent a higher priority request from colliding with requests of lower priority. Moreover, in PPS, the granted bandwidth for lower priority requests can be preempted by the waiting request with higher priority. This will speedup the channel capture by priority data. The efficiency of PPS is investigated by simulations. Simulation results show that by adopting INs with PPS to be an agent for subscribers can not only shorten the collision resolving period but also minimize the average request delay of priority data.

KEY WORDS
Algorithm, CATV, CSMA/CD, HFC, MAC, priority.

1 Introduction

Since in the last decade, the needing of real-time multimedia applications, such as video-on-demand, high quality video conference, is increasing far beyond our expectation. Thus, the contemporary backbone networks are expected to provide priority accesses. The hybrid fiber-coax (HFC) architecture, which has become the standard in cable-television (CATV) industry [2, 3], is one of famous candidates. These CATV proprietors are interested in offering communications as well as on-demand services on this infrastructure.

The HFC architecture is considered as a bi-directional broadband communication infrastructure. It is constructed by optical fibers and coaxial wires. A group of thousand subscribers (also named as stations in this paper) are served by a fiber that comes from the headend controller (HC) to a fiber node (FN). Signals are transmitted electrically from FN to home by coaxial cable through some amplifiers and splitters. Stations attached to the cable transmit and receive signals over different frequencies. The frequency spectrum on the coaxial wire portion of the network is divided into an upstream region (from stations to HC) and a downstream region (from HC to stations). Data rates on the upstream and downstream channels are approximately 3 Mbps and 30 Mbps, respectively. Synchronization at the physical layer is also being considered to ensure that all subscribers have a common time reference.

The upstream channel is divided by HC into fixed minislots which are allocated to stations for requesting and transferring information. At any time, a station transmits data on one upstream channel and receives data on one downstream channel. Each upstream channel is a multi-access channel, and collision occurs when multiple stations transmit simultaneously. On the other hand, all downstream channels are collision-free. Access to the upstream channel is a two-step process. At first, the HC allocates a lot of *request minislots* (RMSs) and informs stations to send requests in these RMSs if they have data to transmit. In other words, a station wants to transmit data on the upstream channel, it needs first send a bandwidth request to the HC. If more than one station transmits a request at the same RMS, these requests collide and a *generic collision resolution algorithm* (G-CRA) is activated by HC to ensure successful retransmission of the requests. Since users cannot listen to the upstream channel, collisions are unable to be detected by stations, and therefore the collision detection is done by HC. Once HC derives the reservation result, it will notify stations when to transmit data (success case) or when to contend again (collision case). Because of the long propagation delay in HFC network, the throughput will become unacceptable if one adopts inefficient collision resolution mechanisms. Thus, some collision resolution mechanisms have been proposed and scheme like ternary tree algorithm was considered in the standard [3].

Another issue in HFC network is how to provide multiple priority levels. In paper [1], authors suggested a simple scheme that can support priorities during contention resolution for tree-search (stack) contention-resolution algorithms. To do this, they proposed a new contention frame structure for IEEE 802.14 protocol. Several RMSs at the beginning of the frame are converted for exclusive use by priority requests/packets. Each of these RMSs, referred to as a *priority newcomer access* (PNA) slot, correspond to a single priority level. The HC identifies a PNA slot with a negative *request queue* (RQ) value, where the RQ value $-N$ is reserved for priority level N. Note that each priority level can send requests to the HC without interference from the other priorities. Basically, this scheme is designed to make sure that higher priority requests are never blocked from requests of lower priority; however, this scheme does not guarantee the high priority data will be served first. This priority scheme may lead to a lower priority request getting bandwidth faster than a higher priority request in a station. This is because that a group of consecutive PNA slots in a contention frame probably contains multiple priorities. Stations having different priority requests are permitted to transmit them onto corresponding PNA slots in a frame. Consequently, the channel access may not obey the priority order. Besides, the proposed priority scheme does not consider how to reduce the number of requests in the contention phase. In general cases, it may still spend a considerable time to resolve collisions of high priority requests as conventional protocol does.

In this paper, we will propose a simple and efficient preemptive priority scheme (PPS), which is based on the frame structure as introduced in [1], to solve the drawbacks of conventional priority scheme. There are two basic concepts in the proposed PPS: (1) The PPS permits a higher priority request to preempt the allocated bandwidth for lower priority data in a station. (2) Based on concept (1), the PPS only allows stations to simultaneously transmit the other lower priority requests with 'sufficient' bandwidth requirement to speedup the process of capturing bandwidth by priority data. The 'sufficient' bandwidth requirement is defined as the amount bandwidth requirement of one or many requests is more than the bandwidth requirement of any higher priority request. Once any request successes in contention, the highest priority data will get the bandwidth immediately as described in (1). Since the PPS does not modify the priority reservation/contention scheme in [1], it is still compatible with the conventional approach. In a word, the proposed PPS will not only improve the bandwidth utilization but also support the multi-priority access in HFC networks.

2 Multi-priority Access Scheme

Fig. 1 shows a simple frame format with multiple priorities. The new frame format was proposed in [1]. Several RMSs at the beginning of the frame are converted for exclusive use by priority packets. Each of these RMSs, referred to as a *priority newcomer access* (PNA) slot, correspond to a single priority level. The HC identifies a PNA slot with a negative RQ value, where the RQ value $-N$ is reserved for priority level N. A larger priority index indicates a higher priority level. (We assume the lowest priority level is 0.) For instance, an RQ value of -2 indicates that the slot is reserved for priority level 2. With PNA slots, each priority level request can be transmitted without interference from other priorities. Thus, when higher priority stations transmit requests, they are never disturbed from requests of lower priority.

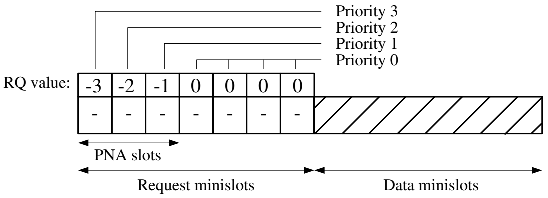


Figure 1. Priority frame format.

2.1 Example of Priority Collision Resolution

Fig. 2 shows an example of the collision resolution process in HFC network with five stations, labelled from A to E. Taking Fig. 2 for example, we assume the HFC network supports four priority levels 0, 1, 2, and 3, where priority levels "3" and "0" are the highest and the lowest priority levels, respectively. The bandwidth requested table of every station (one row for each station) is showed on the left-hand side in Fig. 2. Each entry in this table indicates the total number of requested bandwidth units (could be measured in time slots) of a priority level in a station. For example, station A needs 3 bandwidth units for the packet(s) of priority level 3, 1 bandwidth unit for packet(s) of priority level 2, 1 bandwidth unit for the packet(s) of priority level 2 and 2 bandwidth units for packet(s) of priority level 0. We note that every station only need to maintain its own bandwidth requests as a distributed protocol.

The center of Fig. 2(a) is the contention frame that carries the transmitted requests from five stations. This frame contains 7 RMSs and an unspecified number of DMSs if any. Assume that the system has no previous collisions needed to be resolved, the HC will set the RQ values in the priority frame as shown in Fig. 1. Recall that a negative RQ value $-N$ designates the RMSs as a PNA slot of priority level N. The first three RMSs with RQ values -3, -2, -1 are PNA slots for priority levels 3, 2 and 1, respectively. The remaining PNA slots are assigned a priority level of 0 (i.e., the lowest priority level).

For simplicity, we let An denote the issued request of priority level n by station A. In the first contention frame (also denoted as Frame 1 for simplicity), shown in Fig. 2(a), stations A, C, D, and E transmit requests of priority level 3 (A3, C3, D3 and E3) simultaneously. It is evident that collision occurs since they transmit their requests in the same PNA slot (with RQ = -3). Similarly, requests of priority levels 2 and 1 also collide with the others. For priority level 0, stations B and E have randomly selected different minislots with RQ = 0, and therefore each of them transmits a successful bandwidth request. On the other hand, stations A and C, that transmit requests of priority 0 in the same minislot, also need resolve the collision.

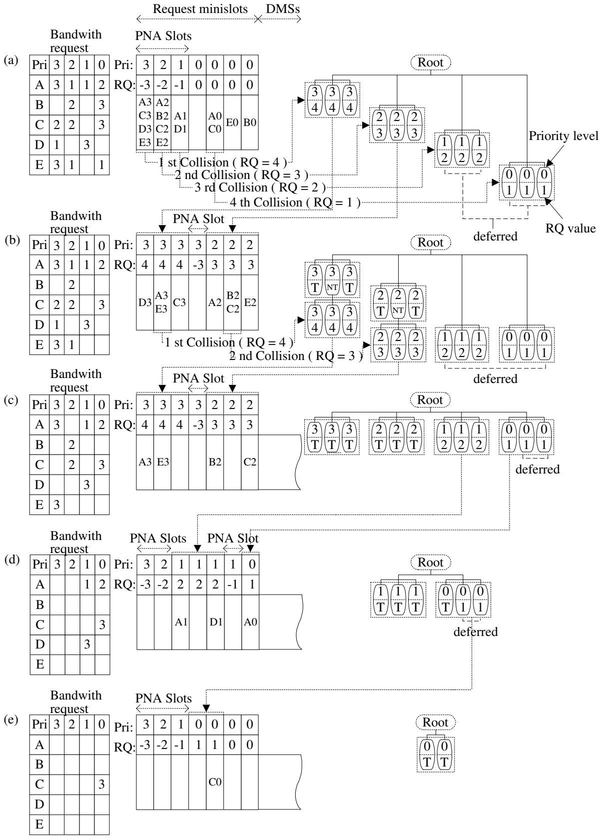


Figure 2. Example of priority collision resolution.

The right-hand side of Fig. 2(a) depicts the corresponding collision tree after the frame (Frame 1) arrived at the HC. (In this example, we assume the collision tree is empty before this frame.) For each collision, a group of three nodes has been added to the collision tree according to the ternary tree algorithm, and these three nodes are labelled with a priority index and an RQ value of the collision. The priority index of a node is identical to the priority index of the PNA minislot where the collision occurred. The RQ values are set as in the uni-priority case, that is, the RQ value is incremented for each collision. In other words, after building and labelling the collision tree, the HC refers the collision tree to assign RQ values to the actual minislots of the RMSs in the next contention frame. Recall that each collision is split across three new minislots and each minislot with the same priority as the collided minislot. In the next frame, some PNA minislots are particularly allocated to provide newcomers of higher priority if there still has enough space. Such PNA minislot comes following the same priority level collision resolution minislots. For example, Fig. 2(b) shows that the Frame 2 contains a new PNA minislot with priority 3 (with RQ = -3) which is posited between RMSs of priority levels 3 and 2.

The HC is responsible for sending feedback message (according to the RMSs' status in previous frame) on the downstream channel to notify collided stations. The feedback message mainly contains the RQ values assigned to the collisions. Thus, requests A3, C3, D3, and E3 are assigned to access the following RMSs with RQ = 4, requests A2, B2, C2, and E2 are assigned to access RMSs with RQ = 3, requests A1 and D1 are assigned to access RMSs with RQ = 2, and requests A0 and C0 are assigned to access RMSs with RQ = 1. The RQs of the first three RMSs are assigned as 4, the RQ of the forth minislot is reserved for PNA with priority level 3 (i.e., RQ = -3) as mentioned above, the remaining minislots (5–7) are assigned as RMSs with RQ = 3. In this step, the number of non-terminated leaf nodes (=12) is larger than the number of available RMSs (=7). Thus, six collided requests (with RQ = 2 and 1) will be deferred until next frame.

In the second contention frame (Frame 2), requests A3, C3, D3, and E3 could select any of the RMSs with RQ = 4. Here, we assume requests C3 and D3 respectively selects the first RMS and the third RMS. Since no other stations select the same minislot, their requests are successful. On the contrary, requests A3 and E3 both collide in the second RMS and need to be resolved again. The forth PNA minislot is open for new request of priority 3 with RQ = -3 and there is no request at this moment. Requests B2 and C2 also collide in the sixth RMS and requests A2 and E2 successfully transmit bandwidth requests on the fifth RMS and the seventh RMS respectively.

The right-hand side of Fig. 2(b) depicts the derived collision tree after Frame 2 arrived at the HC. Any leaf node that corresponds to a minislot which does not contain collision is considered as *terminated* node (labelled as "T") and eliminated from the tree. Leaf node that contains a collision is considered *not terminated* node (labelled as "NT") and it will obtain three children nodes. Again, by Frame 2, the RQ values assigned to the two collisions are RQ = 4 for the first collision (which is caused by requests A3 and E3) and RQ = 3 for the second collision (which is caused by requests B2 and C2). Since there are still not enough RMSs in the next frame to accommodate all minislots needed for collision resolution, the collided requests (with RQ = 2 and RQ = 1) will be further deferred.

The RQ values for the RMSs in Frame 3, as shown in Fig. 2(c), are assigned according to the labels of the leaf nodes in the collision tree: minislots 1–3 are assigned RQ = 4, minislot 4 is assigned RQ = -3 (priority level 3), and the remaining minislots 5–7 are assigned RQ = 3. Fig. 2(c) shows that Frame 3 has no collision. Thus, all nodes of the collision tree, except the six nodes with RQ = 2 and 1 for the deferred requests, are terminated. Fig. 2(d) shows that Frame 4 has no collision. We note that PNA minislots with RQ = -3 and -2 are respectively allocated in the first and the second minislots in Frame 4 due to the priority is higher than deferred nodes' (RQ = 2) priority. Similarly, a PNA minislot with RQ = -1 is allocated in the sixth minislot, which is precedent of the RMS with RQ = 1 (priority 0) allocated in the seventh minislot. We assume station C selects a random number, that is greater than one, for the remaining request C0. Thus, station C transmits request C0 in the next frame Frame 5. Fig. 2(e)shows that one of the RMSs with RQ = 1 is accessed by request C0. After then, all leaf nodes are terminated, which implies that all collisions are resolved.

As mentioned before, the priority scheme may lead to lower priority packets getting bandwidth faster than higher priority packets in station. In the above example, we can find that requests E0, B0, E2 and A2, that with a lower priority level, get the bandwidth earlier than high priority requests. This implies that the priority scheme in [1] is not a perfect priority scheme for the IEEE 802.14 HFC networks. Therefore, it is desired to deign a priority scheme to enhance the access scheme to guarantee the priority access. In the followig section, we will propose the preemptive priority scheme (PPS), which is still compatible with the conventional approach/protocol, to overcome the potential problem.

3 The Preemptive Priority Scheme (PPS)

To make sure the network to serve the high priority data earlier than low priority data in a station, the simple way is to prohibit a station from transmitting lower priority requests if there is any waiting higher priority request. Obviously, the drawback of this approach is the starting time of contentions of lower priority requests will be delayed until all higher priority requests are resolved. Consequently, the time period for resolving all contentions of all requests will be longer than traditional scheme. Hence, an efficient priority scheme should guarantee priority access meanwhile minimizing the contention resolving period.

The proposed PPS still allows a station to transmit different priority requests at the same time as the scheme proposed in [1]. The difference is that the PPS permits a higher priority data to preempt/use the bandwidth allocated for lower priority data in a station. However, the preemption feature still can not guarantee the priority access. This is resulted from a frame may contain different priority request minislots. Thus, the way of reducing the possibility of lower priority data overbearing higher priority data is to limit the number of transmitted lower priority requests in a frame. If we just prohibit stations from transmitting lower priority requests, the bandwidth preemption will never happen. So, it is a tradeoff between the guarantee of priority access and the speed of priority data capturing bandwidth. Thus we need a method to decide the adequate low priority requests to transmit without scarifying the preemption property and without violating the priority access order.

In this section, we propose the priority reservation algorithm (PRA) for stations to determine the proper number of requests to be sent in a contention frame. Excepting the highest priority request, in PRA, the basic constrain of issuing a request with a lower priority is that the accumulative bandwidth requirement from a number of consecutive priority requests excesses the bandwidth requirement of a specific higher priority request. Such transmittable request is named as ‘privileged’ request. Therefore, the first privileged request exists only when the accumulative bandwidth requirement from a number of consecutive lower priority requests is larger than the bandwidth requirement of the highest priority request. The PRA incurs a new problem: since the privileged request may include the bandwidth requirements from different priority levels, what priority level should be associated with the privileged request. By considering the overall priority access, we suggest to associate the highest priority level among all gathered requests with this privileged request. Once the first privileged request is found, the PRA tries to find the second privileged request from its local bandwidth requested table. In stead of comparing with the highest priority request’s bandwidth requirement, the amount of requests bandwidth of the second privileged request must be larger than that of the previous privileged request. The recursion process is repeated until all privileged requests are selected. As a result, for each contention run, several privileged requests may be issued from stations. Once this privileged request has succeed in contention, its bandwidth will immediately contribute for the highest priority data. In the situation that both high priority request and the privileged request are successful simultaneously, all data packets of successful requests will be served.

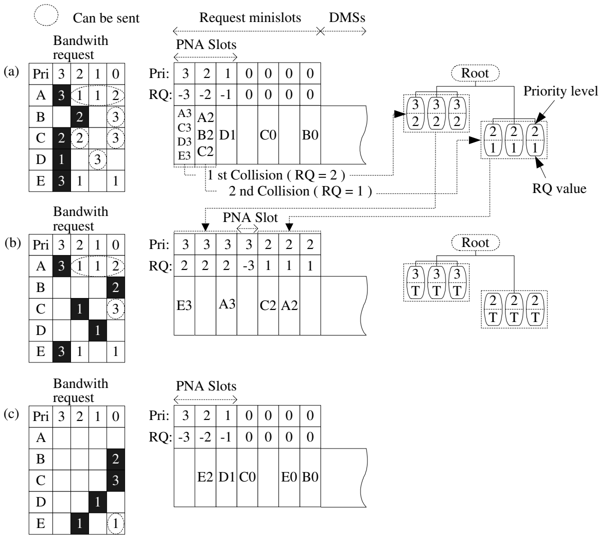


Figure 3. Priority collision resolution by PRA algorithm.

3.1 Example of PPS for Priority Access

For the sake of comparison, we consider the example shown in Fig. 2 again. All bandwidth requests of station are the same as the previous example shown in Section 2.1. The left-hand side of Fig. 3 is the bandwidth requested table that contains the needed bandwidth units of four priority levels in five stations. At first, the PRA algorithm selects the request with the highest priority in each station (they are colored in the bandwidth requested table). That is, stations A, B, C, D and E will send requests A3, B2, C3, D3 and E3 respectively. And then, the PRA algorithm will find the privileged requests if any. For station A, the summation of requested bandwidth units of priorities 2, 1, and 0 is 4 (which are circled by dashed line) and is larger than 3 bandwidth requirement of request A3. Therefore, station A will also send the privileged request A2 with 4 bandwidth units requirement to HC. Similarly, stations B, C and D will transmit requests B0, C2, C0 and D1 by PRA algorithm. On the other hand, station E can not find any privileged request; hence, it only contents resource by its request E3.

The contention (in Frame 1) is shown in the center of Fig. 3(a). All newcomer requests A3, C3, D3, and E3 are transmitted in the same PNA minislot with RQ = -3 and requests A2, B2, and C2 are transmitted in the same PNA minislot with RQ = -2. These two collisions will be detected by HC. On the other hand, requests D1, C0 and B0 success in this round because they access different PNA minislots. The corresponding collision tree for Frame 1 is showed on the right-hand side of Fig. 3(a). For each collision, three new nodes are created, and the nodes are labelled with an priority index and an RQ value. After building and labelling the collision tree, the HC refers it to assign RQ values to RMSs in the next frame. The nodes for the collision occurring at RQ = -3 are labelled with an priority index 3 and an RQ value 2. Another three nodes for the collision occurring at RQ = -2 are labelled with an priority index 2 and an RQ value 1.

In Fig. 3(b), the bandwidth requested table is obviously sparser than original table. Several high priority requests are cleared because that high priority data preempts the bandwidth of low priority data by PPS. For example, when station B obtains three bandwidth units by request B0, it will allocate two bandwidth units to priority 2 (request B2) and allocate the remaining bandwidth (=1) to priority 0. Thus, the bandwidth requirement of priority 0 in station B reduced to 2. Station C allocates the obtained 3 bandwidth units by request C0 to priorities 3 and 2. After then, request of priority 2 in station C still needs 1 bandwidth units. Similarly, station D gets 3 bandwidth units by request D1 and it allocates 1 bandwidth unit to priority 3 and 2 bandwidth units for priority 1.

The center of Fig. 3(b) shows the result of the RQ value assignment for Frame 2. The first three RMSs are assigned RQ = 2, the forth minislot is opened for PNA minislot (priority level 3). The remaining RMSs are assigned RQ = 1 for priority level 2. In Frame 2, we assume requests E3 and A3 select different RMSs with RQ = 2, and requests C2 and A2 also select different RMSs with RQ = 1. Therefore, all transmitted requests are success in Frame 2. In this case, we note that station C will send request C2 since the bandwidth requirement of priority 2 is not cleared.

The remaining bandwidth requirements are shown in the left-hand side of Fig. 3(c). The priority and RQ value assignment for Frame 3, shown in Fig. 3(c), is directly obtained from the collision tree of Fig. 3(b). Since there is no collision in Frame 3, therefore all stations complete their request transmissions and return to the idle state. We can see that the request E2 is deferred until Frame 3 since it is not the privileged request. We also remind that station D transmits request D1 in Frame 3 since it does not occur collision in Frame 1 and therefore waits the PNA minislots with RQ = -1 to transmit the request for bandwidth. This example illustrates that the proposed PRA algorithm is capable of reducing the collision resolving period (reduces 2 collision resolution rounds in this example) as well as maintaining priority access.

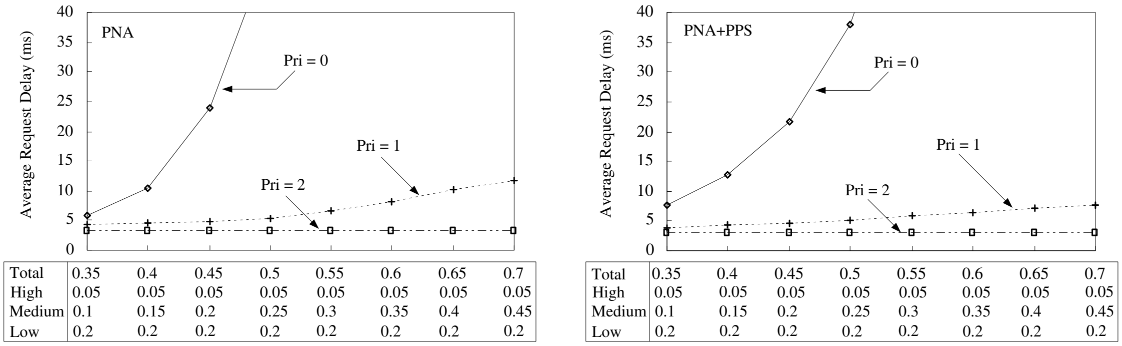
Table 1. System Parameters in Simulations

Simulation Parameter	Normal Values
Total simulation time	10 sec
Distance from nearest/farthest station to HC	25/80 km
Upstream data transmission rates (only one upstream channel is used)	3 Mbps
Propagation delay	5 ms/km for coax and fiber
Data slot size Payload in a data slot	64 bytes 48 bytes
RMS size	16 bytes
DMS/RMS size ratio	4:1
Frame size	52 minislots
Size of RMSs	Fixed 18 slots
Round trip	1 Frame time
Maximum request size	32 data slots
Guard-band and preamble between transmissions from different stations	Duration of 5 bytes
Headend processing delay	1 ms

4 Performance Evaluation and Results

In simulations, we assume the HFC network supports 3 priority levels where priorities 2 and 0 are the highest and the lowest priority levels respectively. In each simulation run, we measure and investigate the *average access delay* (*AAD*) of requests of different priority levels. The access delay of a request is the time interval between the time of request successfully reaches the HC and the time the request arrives at the station. The measured access delay does not include delays that are incurred after the successful transmission of a request, i.e., scheduling delay of the HC and transmission time of data slots.

The detailed configuration and system parameters for the HFC network are shown in Table 1. We assume that there are M stations in the HFC network and the numbers of stations of priority levels 2, 1 and 0 are denoted as M_2, M_1 and M_0 respectively. For each priority level i, the packet arrival rate of station is a Poisson distribution with a mean σ_i. The packet length is an exponential distribution with a mean of L time slots. The *station load* for each station of priority level i (denoted as SL_i) can be defined as $SL_i = \sigma_i \times L$. Hence, the network load (denoted as Λ) can



Total	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7
High	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
Medium	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45
Low	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2

(a) Traffic Load (% of Upstream Capacity)

Total	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7
High	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
Medium	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45
Low	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2

(b) Traffic Load (% of Upstream Capacity)

Figure 4. Comparisons of the average access delays of different priority levels under different network load. (a) *PNA*: network with PNA scheme. (b) *PNA+PPS*: network with PNA scheme and PPS scheme (without INs).

be derived as $\Lambda = \sum_{i=0}^{2} M_i \times SL_i$.

Since the *AAD* is effected by the traffic load, we consider the network load Λ varies from 0.35 to 0.7 in a step of 0.05. In order to further investigate how the *AAD* of a priority is influenced by other priority requests, the percentages of the highest and the lowest priority traffic load are fixed as 5% and 20% of network loaded. We also assign M_2=20, M_1=80 and M_0=100. For comparisons, we investigate the Corner's priority system [1] (denoted as *PNA* scheme) and the effectiveness of the proposed PPS with PNA (denoted as *PNA+PPS* scheme).

4.1 Simulation Results

Fig. 4(a) shows the derived *AAD* of requests of each priority level by PNA scheme. The *AAD* of requests of the highest priority level is well controlled by PNA scheme even under different network loads. This means the PNA scheme indeed provides the priority access for HFC network. From this figure, we can also find that this advantage is derived by scarifying the access delay of the lower priority requests. Even when the network load is only about 0.5, requests of priority 0 will suffer a long access delay (over 50 ms). The major reason is that the transmission of low priority requests will be deferred until all collisions on high priority requests are being resolved successfully.

Fig. 4(b) illustrates the obtained *AAD* of requests of different priority levels by PNA scheme with proposed PPS scheme. We can see that the *AAD* of the highest priority requests is almost the same as that of pure PNA scheme; however, the *AAD*s of the other two priority levels are obviously smaller than that shows in Fig. 4(a). The *AAD* of requests of priority 0 when Λ=0.5 in PNA scheme and in *PNA+PRA* scheme are 51ms and 38ms respectively. The PPS scheme derives about 25% *AAD* improvement. Moreover, the *AAD* of requests of priority 1 when Λ=0.7 in PNA scheme and in *PNA+PRA* scheme are about 11.8ms and 7.5ms respectively. The *AAD* improvement is about 36%. Such significant improvements are resulted from the priority preemption process and the transmissions of privileged requests reduce the the number of contending requests. Consequently, the access delay of a low priority request will also be reduced.

5 Conclusion

This paper proposed a preemptive priority scheme (PPS) with the priority reservation algorithm (PRA) for IEEE 802.14 HFC networks. The proposed PPS can easily derive a better average access delays of requests of all priority levels than that of the priority scheme proposed in [1]. To consider the implementation cost, the designed PPS only slightly modifies the conventional transmission scheme of HFC networks. The proposed PPS can be easily performed in stations and the PRA algorithm is very simple to be development. This result encourages us the PPS with PRA algorithm is practical for supporting priority access in the IEEE 802.14 HFC network.

References

[1] M.D. Corner, J. Liebeherr, N. Golmie, C. Bisdikian, and D.H. Su, A priority scheme for the IEEE 802.14 MAC protocol for hybrid fiber-coax networks, *IEEE/ACM Trans. Networking*, *8*(2), 2000, 200–211.

[2] Y.-D. Lin, On IEEE 802.14 Medium Access Control Protocol, *IEEE Commun. Surveys*, *1*(1), 1998, 2–9.

[3] IEEE 802.14 Working Group, *IEEE project 802.14/a draft 3 revision 3: Cable-TV access method and physical layer specification (unapproved)*, I. Reede, M. Brandt, and J. Karaoguz, eds., IEEE 802.14 Project Draft Specification, NY, 1998.

[4] S.-T. Sheu and M.-H. Chen, A New Network Architecture with Intelligence Node (IN) to Enhance IEEE 802.14 HFC Networks, *IEEE Trans. Broadcasting*, *45*(3), 1999, 308–317.