PAPER

A Preemptive Priority Scheme for Collision Resolution in HFC Networks

Jenhui CHEN[†a)], *Member*, **Shiann-Tsong SHEU**[††], *and* **Sheng-Kun SHEN**[†††], *Nonmembers*

SUMMARY 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 [12]. 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, cable modem, HFC, MAC, priority*

1. Introduction

In the last decade, the needing of e-business interactive services and real-time multimedia applications, such as video-on-demand, high quality videophone, video conference, and high-speed Internet access, is increasing far beyond our expectation. Thus, the contemporary backbone networks are expected to provide multiple priority levels. The hybrid fiber coax (HFC) architecture [11], [13], in which a fiber is used to transport subscriber's multiplexed signals to a group of subscribers, has become the standard in cable-television (CATV) industry [7], [9]. Furthermore, a data-over-cable services interface specifications (DOCSIS) [4] has also been proposed to deploy a high-speed packet-based communications system on CATV infrastructure. 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. The HFC network is constructed by optical fibers and coaxial wires. The coaxial wire portion of the network extends from a fiber-optic interconnected node to the subscribers' home, as shown in Fig. 1. 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). The upstream spectrum is in the range from 5 to 42 MHz with variable size channels typically from 1 to 3 MHz. The downstream spectrum typically ranges from 50 to 860 MHz, divided into channels of fixed width, e.g., 6 MHz in North America and 8 MHz in Europe. Data rates on the channels are approximately 3 Mb/s and 30 Mb/s in the upstream and downstream channels, respectively. Synchronization at the physical layer is also being considered to ensure that all subscribers have a common time reference.

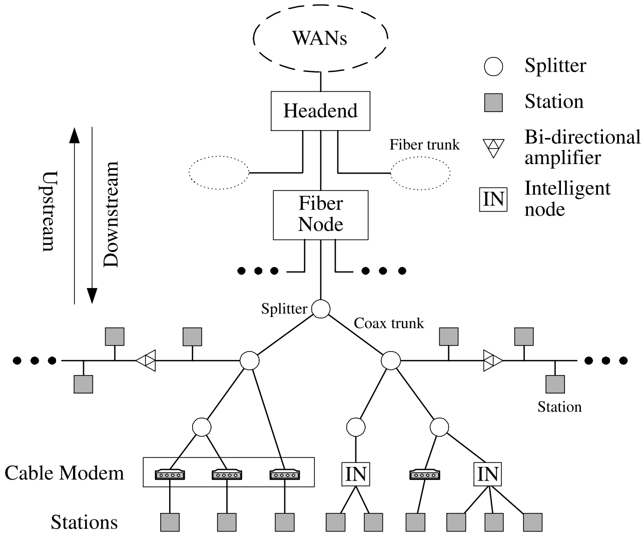


Fig. 1 An example of HFC architecture with intelligent nodes (IN).

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 only on one upstream channel and receives data only 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

---

Manuscript received January 22, 2004.
Manuscript revised May 14, 2004.

[†] The author is with the Department of Computer Science and Information Engineering, Chang Gung University, Tao-Yuan, Taiwan 333, Republic of China.

[††] The author is with the Department of Electrical Engineering, Tamkang University, Tamshui, Taiwan 251, Republic of China.

[†††] The author is with the Department of Communications Engineering, National Tsing Hua University, Hsinchu, Taiwan 300, Republic of China.

a) E-mail: jhchen@mail.cgu.edu.tw

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. This also implies that traditional carrier sense multiple access with collision detection (CSMA/CD) protocol is not suitable in HFC network scenarios. 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 [7]. Some performance issues for contention resolution algorithms are studied in [6], [14].

In our previous work [12], we had proposed the intelligent node (IN) to shorten the collision resolving period in HFC networks. The INs are placed between the headend controller (HC) and stations. The IN will stand for its downstream stations to contend for the demand resources by issuing a single request message with the summed bandwidth requirement onto one of RMS after collecting requests from downstream stations. Upon the IN gets the granted message from HC, it will inform its downstream stations when to access upstream channel in a collision-free manner. Based on this concept, the collision probability and the collision resolving period can be reduced by INs obviously. In practice, we can place an IN in a building and all stations in this building entrust IN to contend resources. When the number of active stations behind an IN is more than one, the collision probability definitely will be smaller than traditional HFC network.

Another issue in HFC network is how to provide multiple priority levels. In paper [5], 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 and has since been incorporated into IEEE 802.14 standard in April 1998. 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 [5], 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 or IN. (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 which is more than the amount bandwidth requirement of any higher priority requests. Once any request successes in contention, the highest priority data will get the bandwidth immediately as described in (1). For the sake of flexibility, the proposed PPS protocol can be implemented in either an IN or a station. This means the designed PPS can be applied in the HFC network with or without INs. Besides, since the PPS does not modify the priority contention/reservation scheme in [5], 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 network.

The remainder of this paper is organized as follows. Section 2 takes an overview of IEEE 802.14 MAC protocol and the multiple priority access scheme proposed in [5]. In Sect. 3, we will briefly describe the architecture of IN. Section 4 presents the proposed preemptive priority scheme (PPS) for IEEE 802.14 HFC networks. In Sect. 5, we give the performance evaluation of PPS and show the impact of it to HFC networks. Finally, some conclusion remarks are given in Sect. 6.

2. The IEEE 802.14 MAC Protocol

In this section, we take an overview of the operations of the IEEE 802.14 MAC protocol and describe the priority contention/reservation scheme in details.

2.1 MAC Operation

Periodically, the HC allocates a number of discrete RMSs

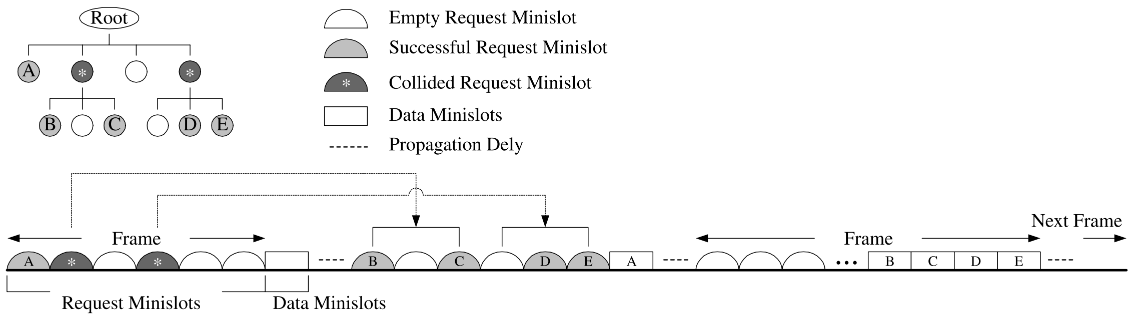


Fig. 2 The Q-ary tree algorithm (Q = 3).

in the upstream channel for stations to transmit request information. The exact timings of RMSs are informed by the RMS grant messages which is generated from HC in the downstream channel. A RMS grant message identifies a number of RMSs divided into groups for different distinct sets of stations which are at various stages of contention resolution. A group of RMSs is allocated for initial contention access. To reserve the transmission resources in the upstream channel, an active station randomly selects an RMS from the group of available RMSs, and then sends a request message onto the selected RMS. In IEEE 802.14 HFC network, collisions will occur in RMSs when more than one station transmits the request onto the same RMS. We note that the data minislots (DMSs) are collision-free and are explicitly allocated to a specific station by the HC.

The HC controls the usage of RMSs by assigning a RQ value to each RMS. A station with a new request desiring to access channel has to obey the *first transmission rule* (*FTR*) [1]. The FTR specifies that the station must wait for a group of RMSs whose RQ is 0 (A RMS with RQ = 0 is called a newcomer RMS). The station then picks a number, say p, between 0 and a range parameter R. If a group of RMSs has less than p minislots with RQ = 0, the station waits for the next cluster of newcomer RMSs and tries again. Otherwise, it transmits the request in the p-th RMS with RQ = 0.

2.2 Generic Collision Resolution Algorithm

If two or more stations transmit requests on the same RMS, the HC executes a generic collision resolution algorithm (G-CRA). In the IEEE 802.14 MAC protocol, the G-CRA is a blocking Q-ary tree algorithm. The operations of the Q-ary tree algorithm is briefly described as follows:

1. The HC allocates some RMSs with assigned RQ values for stations to send their request information.
2. Station having data to transmit randomly selects a RMS and transmits its request message on it.
3. After HC collects all RMSs, it will obtain the contention result. If there is a collision, after a round-trip propagation delay, HC will respectively allocate a number of Q RMSs to stations, which collided in the same RMS.
4. The HC will repeat step 3 until all the collisions are resolved.

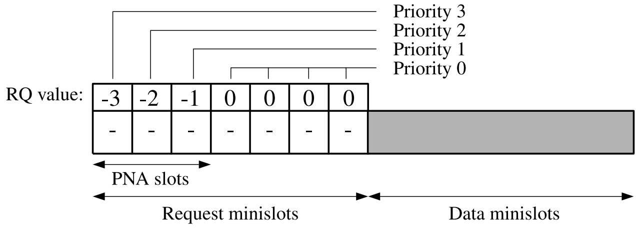


Fig. 3 Priority frame format.

5. Finally, HC will send a DMS Grant Message to inform stations the corresponding DMSs that stations can transmit.

The collision resolution approach can be easily implemented by a stack data structure [10]. The simulation results from [2], [10] show that the ternary (Q = 3) tree algorithm achieves the shortest collision resolution interval and the binary (Q = 2) tree algorithm is close behind.

Taking Fig. 2 for example, at the first time, HC allocates 4 RMSs for stations to contend. Stations B and C collide in the second RMS and stations D and E collide in the last RMS, respectively. Only station A successes because no collision occurs in the first RMS. After a round-trip propagation delay, the HC allocates three new RMSs (in this case, Q = 3) for stations B and C and another three new RMSs for stations D and E. In this case, we assume no collision occurs in the second round. As soon as all stations are success in contention, the HC allocates DMSs to these five stations.

2.3 Multi-Priority Access Scheme

Figure 3 shows a simple frame format with multiple priorities. The new frame format was proposed in [5]. 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.

2.4 Example of Priority Collision Resolution

Figure 4 shows an example of the collision resolution process in HFC network with five stations, labeled from A to E. We note that the scheme proposed in [5] doesn't consider the case of stations having different priority packets, but we affirm that their scheme can support multiple priorities as well. Taking Fig. 4 for example, we assume the HFC network supports four priority levels 0, 1, 2, 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. 4. 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.

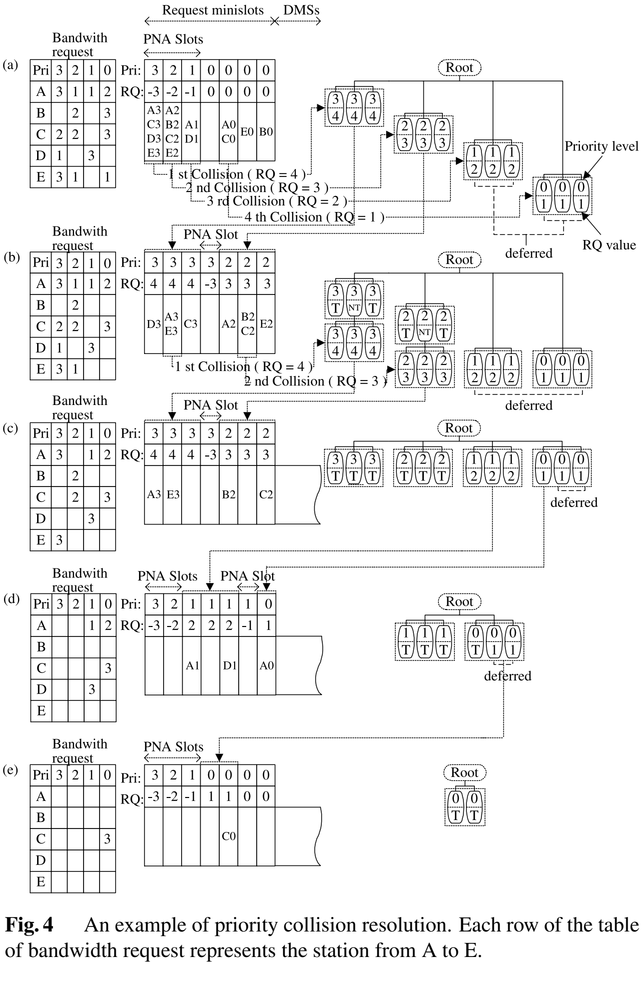


Fig. 4 An example of priority collision resolution. Each row of the table of bandwidth request represents the station from A to E.

The center of Fig. 4(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. 3. 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. 4(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.

The right-hand side of Fig. 4(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 labeled 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. 4(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. 4(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 (labeled as “T”) and eliminated from the tree. Leaf node that contains a collision is considered *not terminated* node (labeled 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. 4(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. Figure 4(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. Figure 4(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. Figure 4(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 [5] 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 section 4, we will propose the preemptive priority scheme (PPS), which is still compatible with the conventional approach/protocol, to overcome the potential problem. Before describing the proposed PPS, we first briefly describe the functionality of intelligent nodes (INs) we had proposed in [12].

3. The Intelligent Nodes (INs)

3.1 The Architecture of INs

Traditionally, a station desiring to transmit data should send the request onto a limited number of RMSs to contend. The performance may be degraded with the increase of the number of stations. This is the major drawback of CATV network to support thousands of stations. In the proposed network architecture in [12], we place some *intelligent nodes* (INs) in the traditional HFC network as shown in Fig. 1. The role of an IN is the agent of a group of stations. From the station’s point of view, the IN can be treated as the HC. If there are several active stations want to transmit data simultaneously, the IN will substitute for sending a single request message with the summation bandwidth. To avoid the collision occurring between stations and IN, we solve this problem by employing a Time Division Multiple Access (TDMA)-based control channel shared by all stations. Obviously, when the number of active stations under an agent IN is more than one, the collision probability will be decreased. After finishing contention, IN will inform its downstream stations when to transmit data by control channel.

When an IN is power on, it must first acquire a downstream channel. If the downstream channel does not contain data stream, it should select another downstream channel. After acquiring a downstream channel, the procedure timing acquiring and ranging is the most important step in the initial state, which determines the *round-trip correction* (RTC) parameter. This is done by the HC periodically inviting newcomers by sending ranging invitation message through the downstream channel. After these procedures, the IN would get the RTC parameter. The RTC value equals to the difference between the network’s maximal round-trip propagation delay and the round-trip propagation delay between HC and IN.

In order to accommodate the new network architecture, a station needs to modify the transmission process as two states: convention station (without IN) and agent state (with IN). Now, we will describe the operations of these two states.

Convention State: In this state, station contends channel by itself when it has data to transmit. After contention resolving procedure, it will receive the DMS Grant Message from HC. And then it waits for the DMSs and transmits data on them. If he piggyback function is enabled, the station sets one flag in its packets, as its buffer is not empty. The HC will reserve more DMSs for this station to transmit data in the next period without contending again. This will significantly reduce the number of requests contention in upstream channel.

Agent State: After entering agent state, stations performs the procedures of channel acquiring, timing acquiring and ranging from IN. And then the station registers to the IN. When it has data to transmit, it waits for the RMS Grant Message from IN and sends the request in its unique RMS. (In this state, we employ the time division multiplexing (TDM) approach for station delivering requests to IN in a collision-free manner.) After this step, the station only needs to wait for the DMS Grant Message from IN and transmitting data. The piggyback method is the same as the convention state.

In practice, the IN can be placed in buildings. Stations in the building entrust the IN to contend the resource. It is very feasible by using the INs because we do not modify the traditional architecture. Stations who use the traditional equipment need not change. Only the user equipments using the IN needed to be modified slightly. In the next section, we will propose the preemptive priority scheme (PPS) for IEEE 802.14 HFC networks. This protocol can be performed either in the conventional network architecture or the network architecture with INs.

4. 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 [5]. The difference is that the PPS permits a higher priority data to preempt/use the bandwidth allocated for lower priority data in a station or IN. However, by only performing the preemption 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. Therefore, 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/INs 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/INs. Liking the request issued from IN, the privileged request carries the summation bandwidth of a group of requests from different priorities. 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. Based on this concept, the priority access in station/IN will be preserved and the benefit of priority preemption will be maintained also. In the next subsection, we will describe the PRA algorithm in details.

4.1 PRA Algorithm

Without loss of generality, we assume the HFC network supports N priority levels and they are indexed from 0 to $N-1$. Data of priority i must be served before data of priority j if $i > j$ $(0 \leq i, j < N)$. Each station locally maintains a bandwidth requested table which records the accumulated bandwidth requirement for each priority level. Thus, let $\mathcal{R} = \{R_{N-1}, R_{N-2}, \cdots, R_0\}$ denote the set of bandwidth requirements in a station/IN where R_i denotes the total required bandwidth units at priority level i.

Assume set $\mathcal{T} = \{T_1, T_2, \cdots, T_m\}$ denote the transmittable requests (including the privileged requests) by PRA where $T_i = (p_i, l_i, r_i)$ means the i-th request with priority level p_i needs r_i bandwidth units[†] and the lowest priority level among gathered requests is l_i. Initially, $\mathcal{T} = \phi$. Upon the RMS Grant message arrival, for each contention frame, the PRA first finds the highest priority level (say k), whose bandwidth requirement is not zero, from set $\mathcal{R}$. (That is, $R_j = 0, \forall j > k$.) Then, PRA adds element $T_1 = (p_1, l_i, r_1) = (k, k, R_k)$ into set $\mathcal{T}$. After then, the PRA will try to add the first privileged request T_2, whose accumulated bandwidth requirement with priority lower than l_1 (i.e., k) is larger than r_1 ($= R_k$), into set $\mathcal{T}$. The lowest priority level (l_2) of requests gathered in the first privileged request T_2 can be determined as following:

$$l_2 = \max\left\{ j \,\middle|\, \sum_{i=j}^{l_1-1} R_i > r_1, j \geq 0 \right\}. \tag{1}$$

Once l_2 is derived, the priority level p_2 of the first selected privileged request can be also determined. We have

$$p_2 = \max\left\{ j \,\middle|\, R_j > 0, l_2 \leq j < l_1 \right\}. \tag{2}$$

Consequently, the amount of bandwidth requirement for the privileged request T_2 can be derived as follows:

$$r_2 = \sum_{i=l_2}^{l_1-1} R_i. \tag{3}$$

The priority level p_m, the lowest priority level l_m and the amount of requested bandwidth r_m of T_m ($m > 1$) can be derived as following:

$$l_m = \max\left\{ j \,\middle|\, \sum_{i=j}^{l_{m-1}-1} R_i > r_{m-1}, j \geq 0 \right\}. \tag{4}$$

$$p_m = \max\left\{ j \,\middle|\, R_j > 0, l_m \leq j < l_{m-1} \right\}. \tag{5}$$

$$r_m = \sum_{i=l_m}^{l_{m-1}-1} R_i. \tag{6}$$

The detailed PRA algorithm is shown in Fig. 5. The time complexity of the PRA is $O(N)$ where N is the number of priority levels in network. This implies that the PRA is easy to be implemented. Combining it with IN nodes, the PRA algorithm provides an efficient and rapid way to allocate bandwidth for high priority data since the bandwidth is collected together and reassigned for packets according to their priority order. Figure 6 illustrates the new state machine of stations.

Notice that the proposed PRA is independent of industrial standards since the PRA utilizes the capacity of priority to improve its performance. Kuo et al. [8] proposed a dynamic backoff scheme for supporting multiple priority traffic over DOCSIS cable networks. According to this algorithm, the PRA could be applied in the DOCSIS cable networks as well.

```
PRA–PRIORITY-RESERVATION-ALGORITHM($\Psi, \varphi$)

Forward scan of all request bandwidth $\mathcal{R} \in \Psi$
  Denote the request bandwidth with highest priority
  level as $\mathcal{S}_H$
  $\Psi \Leftarrow \Psi - \mathcal{S}_H$
  $\varphi \Leftarrow \mathcal{S}_H$
  $i \leftarrow$ highest priority level$-1$
  Repeat until $\Psi = \phi$
    if $\mathcal{S}_t = \varepsilon$
      $t \leftarrow i$
    $\mathcal{S}_t \Leftarrow \mathcal{S}_t + \mathcal{R}_i$
    $\Psi \Leftarrow \Psi - \mathcal{R}_i$
    if $\mathcal{S}_t \geq N_i^-$
      $\varphi \Leftarrow \varphi + \mathcal{S}_t$
      $\mathcal{S}_t \leftarrow \varepsilon$
    else
      $i \leftarrow i - 1$
  end Repeat
Receive quantum of DMS Grant Message $G$ from HC
  Repeat until $G = \phi$
    Pick up the highest priority level $\mathcal{S}_h$ from $\varphi$
    if $\mathcal{S}_h \leq G$
      $G \leftarrow G - \mathcal{S}_h$
      $\varphi \Leftarrow \varphi - \mathcal{S}_h$
    else
      $\mathcal{S}_h \leftarrow \mathcal{S}_h - G$
  end Repeat
```

Fig. 5 Priority reservation allocation (PRA) algorithm.

4.2 Example of PPS for Priority Access

In this section, we would demonstrate how the PPS scheme solves the collision and priority access. For the sake of comparison, we consider the example shown in Fig. 4 again. All bandwidth requests of station are the same as the previous example shown in Sect. 2.4.

The left-hand side of Fig. 7 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 tries to 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. 7(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 min-

---

[†] The amount of bandwidth units is derived from accumulating the bandwidth requirements of requests of different priorities

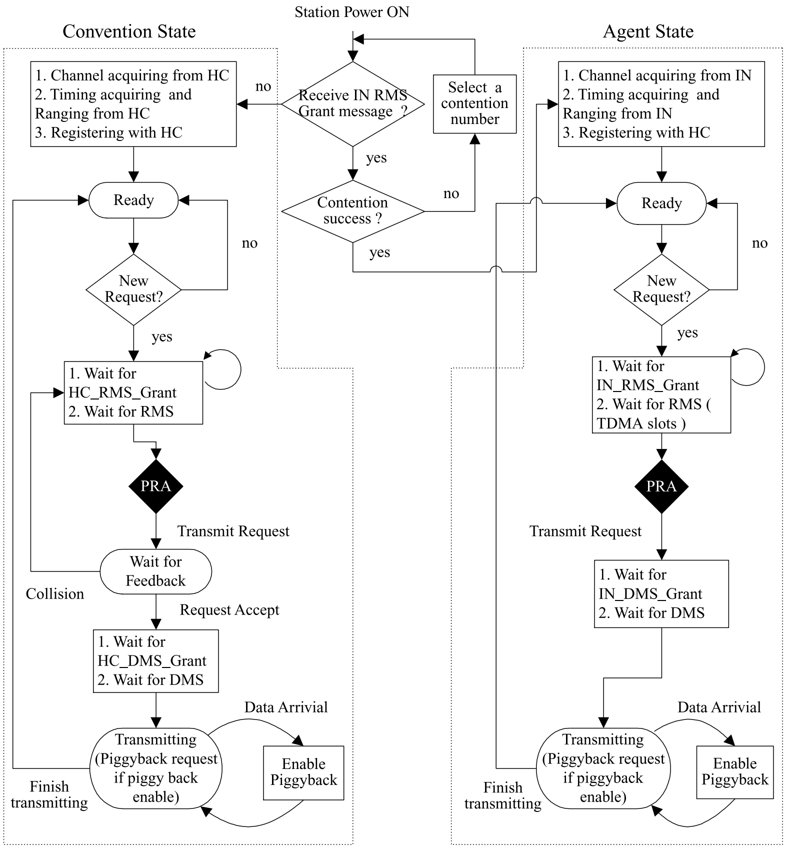


Fig. 6 The state machine of station with PRA algorithm.

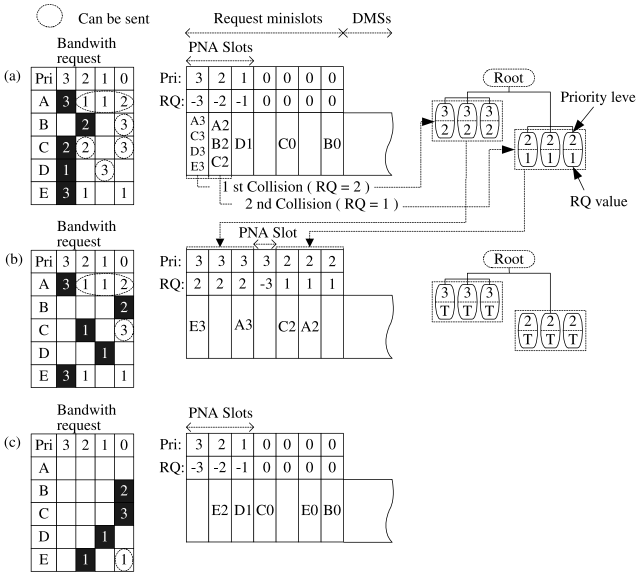


Fig. 7 Priority collision resolution by PRA algorithm.

islot 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. 7(a). For each collision, three new nodes are created, and the nodes are labeled 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 labeled with a priority index 3 and an RQ value 2. Another three nodes for the collision occurring at RQ = −2 are labeled with an priority index 2 and an RQ value 1.

In Fig. 7(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. 7(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

Table 1 Comparison of merits in G-CRA, PNA, and PRA.

	G-CRA	PNA	PRA
Priority capability	no	yes	yes
Various priority levels	no	yes	yes
Priority preemption	no	yes	yes
Aggregation of requests	no	no	yes
Bandwidth sharing	no	no	yes
Faster collision resolution	no	no	yes

Table 2 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 Mb/s
Propagation delay	5 ms/km for coax and fiber
Data slot size	64 bytes
Payload in a data slot	48 bytes
RMS size	16 bytes
DMS/RMS size ratio	4:1
Frame size	52 minislots
Size of RMSs	Fixed 18 minislots
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

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. 7(c). The priority and RQ value assignment for Frame 3, shown in Fig. 7(c), is directly obtained from the collision tree of Fig. 7(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 doesn't 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. In Table 1, we summarize the comparison of merits of early mentioned minislots reservation allocation algorithms. We can see that the proposed PRA provides not only several priorities for choice but aggregation of several priorities in one highest priority of these requests to reduce the contention overhead thus improving the performance of collision resolution in HFC networks.

5. Performance Evaluation and Results

5.1 Simulation Model

In this section, we will demonstrate the performance of our proposed PPS with PRA algorithm for the IEEE 802.14 HFC network. In simulations, we assume the HFC network supports 3 priority levels where priorities 2 and 0 are the highest and lowest priority levels respectively. In each simulation run, we measure and investigate the *average request delay* (ARD) of requests of different priority levels. The request delay is the time interval it takes a transmission request to successfully reach the HC from the time the request arrives at the station. The measured request 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 2. 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

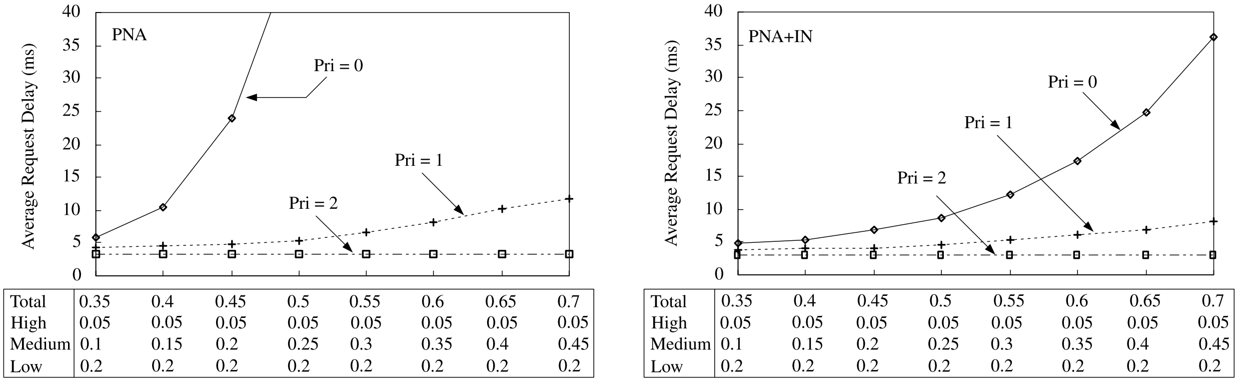
$$SL_i = \sigma_i \times L.$$

Hence, the network load (denoted as Λ) can be derived as

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

Since the ARD 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 ARD 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 $M_0 = 100$. Four different experiments are simulated for comparing the effectiveness of the architecture of IN, the PPS, and the Corner's priority system [5] (for simplicity, we name it as PNA scheme in short). Four simulation environments are listed as follows:

- In Experiment 1 (denoted as *PNA*), we only investigate the efficiency of conventional PNA scheme.
- In Experiment 2 (denoted as *PNA+PRA*), based on PNA scheme, we observe the performance of PPS scheme.
- In Experiment 3 (denoted as *PNA+IN*), the effectiveness of employing INs with PPS is observed.
- In Experiment 4 (denoted as *PNA+IN+PRA*), we investigate the effectiveness of combining INs, PPS, and PNA scheme.

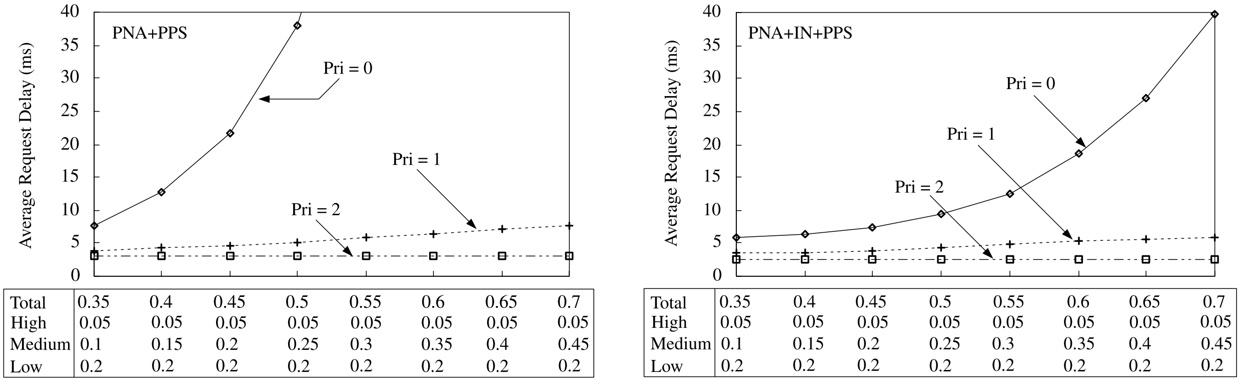


Total	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7
High	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
Medium	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45
Low	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2

(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

(c) 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)

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

(d) Traffic Load (% of Upstream Capacity)

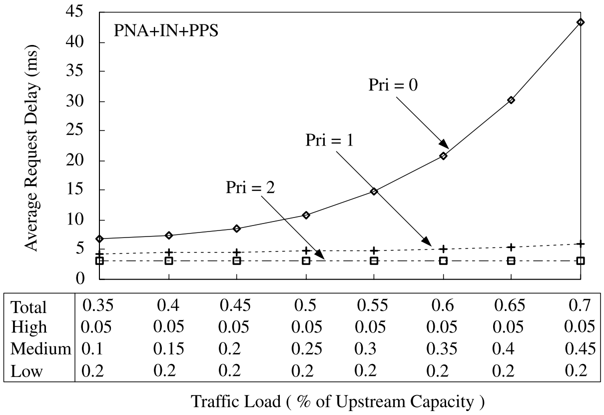
Fig. 8 Comparisons of the average request delays of different priority levels under different network load. (a) *PNA*: network with PNA scheme (without INs and PPS scheme). (b) *PNA+PPS*: network with PNA scheme and PPS scheme (without INs). (c) *PNA+IN*: network with PNA scheme and INs (without PPS scheme). (d) *PNA+IN+PPS*: network with PNA scheme, INs, and PPS scheme.

5.2 Simulation Results

Figure 8(a) shows the derived ARD of requests of each priority level by PNA scheme. The ARD 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.

Figure 8(b) illustrates the obtained ARD of requests of different priority levels by PNA scheme with proposed PPS scheme. We can see that the ARD of the highest priority requests is almost the same as that of pure PNA scheme; however, the ARDs of the other two priority levels are obviously smaller than that shows in Fig. 8(a). The ARD of requests of priority 0 when $\Lambda = 0.5$ in PNA scheme and in *PNA+PRA* scheme are 51 ms and 38 ms respectively. The PPS scheme derives about 25% ARD improvement. Moreover, the ARD of requests of priority 1 when $\Lambda = 0.7$ in PNA scheme and in *PNA+PRA* scheme are about 11.8 ms and 7.5 ms respectively. The ARD 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.

Figure 8(c) shows the effectiveness of network architecture with INs. Recall that the main functionality of INs is to reduce the number of requests in network. The INs will gather all requests from downstream stations and then send the request(s), so that the number of requests can be reduced. Owing to the ARD is proportional with the number of contending requests in network, the ARD will be contra-proportional with the number of INs in network. In Experiment *PNA+IN*, we consider locating 5, 20, and 25 numbers of INs in M_2, M_1, and M_0, respectively. Besides, a number of 200 data stations are equally allocated for INs. (i.e., each IN will handle 4 stations.) From Fig. 8(c), we can see that the ARD of requests of priority 2 is still bounded around 3.5 ms for all kinds of network load under different numbers of INs. As the number of INs decreases (increases), the ARD of low priority requests will become smaller (larger). The phenomena is resulted from the number of requests of stations are reduced as the number of INs. When the num-



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

Fig. 9 The comparison of the average request delays of different priority levels under different network load when applying *PNA+IN+PPS* and adopting 50 INs and 2,000 data stations.

ber of INs in network is reduced as one, we can imagine that the ARD of the highest priority requests will approach zero. Nevertheless, it is impractical that HFC network only adopts one IN for all stations since the communication between the IN and downstream stations is following a TDMA-based protocol. (The request and grant messages from station or IN will be transmitted without contention.) Thus, the maximal number of stations can be handled in a IN is depending on the frame size of the TDMA protocol. Therefore, in practice, the reasonable number of INs in network may be over one hundred if the HFC networks supports thousands of stations.

Figure 8(d) shows the performance of environment *PNA+IN+PPS*. The difference between environments *PNA+IN+PPS* and *PNA+IN* is the INs performs PPS scheme in the former case. We can easily see that the ARD curve of the highest priority level in environments *PNA+IN+PPS* and *PNA+IN* are very close to each other. In fact, the ARD of priority 2 in the environment *PNA+IN+PPS* is about 2.9 ms which is slightly smaller than 3.4 ms in the environment *PNA+IN*. The improvement on ARD of priority 1 by PPS will become more obvious when the traffic load increases. This is because that PPS assembles several low priorities into one high priority to contend the bandwidth, and thus reduces the contention overhead and curtails the time of ARD. This implies that the PPS will obtain significant improvement only when the number of requests is moderate.

In order to reveal the scalability of proposed *PNA+IN+PPS*, we perform a large scale CATV environment in the following simulation. We extend the data stations to 2,000 and keep the original number of INs in our simulation. We can see that, from Fig. 9, the ARD of each priority only increases few milliseconds. The ARD of priority 2 increases from 2.9 ms to 3.1 ms and the lowest priority does not cause longer request delay (from 39.8 ms to 43.3 ms comparison to Fig. 8(d) when traffic load reaches 0.7). This result encourages us to apply both PRA and INs in HFC architecture for a large backbone network.

6. Conclusion

We, in this paper, proposed and investigated a preemptive priority scheme (PPS) with the priority reservation algorithm (PRA) for IEEE 802.14 HFC networks with and without INs. The PPS with PRA is based on the priority scheme proposed in [5]. By only employing PPS or INs can easily derive the better average access delays of requests of all priority levels than that of the priority scheme proposed in [5]. Besides, The combination of INs and PPS will have the potential capability to achieve a higher throughput in ordinary HFC networks. To consider the implementation cost, the designed INs or PPS may slightly modify the conventional architecture and transmission scheme of HFC networks. The cost of implementation is minimized since both IN and PPS take the standardized IEEE 802.14 protocol for resolving collisions. The proposed PPS can be easily performed in either stations or INs (if any) and the PRA algorithm is very simple to be development. This result encourages us to practice the PPS with PRA algorithm for supporting priority access in IEEE 802.14 HFC network.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments.

References

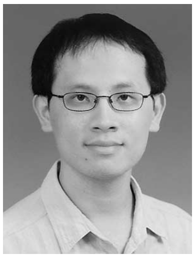
[1] C. Bisdikian, “A review of random access algorithms,” contribution to the IEEE 802.14 WG, no.IEEE 802.14-96/019, Jan. 1996.
[2] C. Bisdikian, B. McNeil, R. Zeisz, and R. Norman, “MLAP: A MAC level access protocol for the HFC 802.14 network,” IEEE Commun. Mag., vol.34, no.3, pp.114–121, March 1996.
[3] C.C. Bisdikian, “Throughput behavior of the n-ary stack algorithm in mobile networks with capture,” Proc. 20th Conf. Local Computer Networks, pp.453–458, 1995.
[4] Cable Television Laboratories, Data-Over-Cable Service Interface Specifications—Radio Frequency Interface Specification version 1.1, July 1999.
[5] 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. Netw., vol.8, no.2, pp.200–211, April 2000.
[6] N. Golmie, Y. Saintillan, and D. Su, “Review of contention resolution algorithms for IEEE 802.14 networks,” IEEE Commun. Surveys, http://www.comsoc.org/pubs/surveys, vol.2, no.1, 1999.
[7] IEEE, “IEEE project 802.14/a draft 3 revision 3: Cable-TV access method and physical layer specification (unapproved),” ed. I. Reede, M. Brandt, and J. Karaoguz, IEEE 802.14 Project Draft Specification, NY, Oct. 1998.
[8] W.-K. Kuo, S. Kumar, and C.-C.J. Kuo, “Improved priority access, bandwidth allocation and traffic scheduling for DOCSIS cable networks,” IEEE Trans. Broadcast., vol.49, no.4, pp.371–382, Dec. 2003.
[9] Y.-D. Lin, “On IEEE 802.14 medium access control protocol,” IEEE Commun. Surveys, vol.1, no.1, pp.2–9, Quarter 1998.
[10] P. Mathys and P. Flajolet, “Q-ary collision resolution algorithm in random access systems with free and blocked channel access,” IEEE Trans. Inf. Theory, vol.IT–31, no.2, pp.217–243, March 1985.

[11] S. Ramanathan and R. Gusella, “Toward management systems for emerging hybrid fiber-coax access networks,” IEEE Netw., vol.9, pp.58–68, Sept./Oct. 1995.

[12] S.-T. Sheu and M.-H. Chen, “A new network architecture with Intelligence Node (IN) to enhance IEEE 802.14 HFC networks,” IEEE Trans. Broadcast., vol.45, no.3, pp.308–317, Sept. 1999.

[13] M. Vecchi, “Broadband networks and services: Architecture and control,” IEEE Commun. Mag., vol.33, no.8, pp.24–32, Aug. 1995.

[14] W.M. Yin and Y.D. Lin, “Statistically optimized minislot allocation for initial and collision resolution in hybrid fiber coaxial networks,” IEEE J. Sel. Areas Commun., vol.18, no.4, pp.1764–1773, Sept. 2000.



Jenhui Chen was born on October 12, 1971 in Taipei, Taiwan, Republic of China. He received the Bachelor’s and Ph.D. degree in Computer Science and Information Engineering (CSIE) from Tamkang University in 1998 and 2003, respectively. In the spring of 2003, he joined the faculty of Computer Science and Information Engineering Department at Chang Gung University and served as the Assistant Professor. He occupies the supervisor of Network Department in the Information Center, Chang Gung University. Dr. Chen once served the reviewer of IEEE Transactions on Wireless Communications, ACM/Kluwer Mobile Networks and Applications (MONET), and Journal of Information Science and Engineering. His main research interests include design, analysis, and implementation of communication and network protocols, wireless networks, milibots, and artificial intelligence. He is a member of ACM and IEEE.

Shiann-Tsong Sheu received his B.S. degree in Applied Mathematics from National Chung Hsing University in 1990, and obtained his Ph.D. degree in Computer Science from National Tsing Hua University in May of 1995. From 1995 to 2002, he was an Associate Professor at the Department of Electrical Engineering, Tamkang University. Since February 2002, he has become a Professor at the Department of Electrical Engineering, Tamkang University. Dr. Sheu received the outstanding young researcher award by the IEEE Communication Society Asia Pacific Board in 2002. His research interests include next-generation wireless communication, WDM networks and intelligent control algorithms.

Sheng-Kun Shen was born in Yun-Lin Hsien, Taiwan on October 28, 1977. He received the B.S. degree in the Department of Electrical Engineering, Tamkang University and M.S. degree in the Department of Communications Engineering, National Tsing Hua University, Taiwan, R.O.C., in 2001 and 2003, respectively. Currently, he is an engineer in Accton Technology Corporation. His current research interests include design, analysis and implementation of network protocols, wireless communications and HFC networks.