(1) Name of workshop: Workshop on Computer Networks

(2) Title: An Adaptive contention period control in HFC Networks

(3) Abstract

As the deployment of the cable TV (CATV) network becomes ubiquitous, the CATV networks have emerged as one of primary technologies to provide broadband access to the home. Due to the natural of CATV network－the radical asymmetric bandwidth, the upstream channel is critical to allocating and scheduling. The objective of this paper is to propose a new adaptive contention period control algorithm in HFC networks to improve the utilization and throughput of the upstream bandwidth. We proposed our method in Section 4 and simulated results in Section 5. Through the simulation, we have shown that in any cases, our adaptive method performs better than MCNS DOCSIS.

(4) Name: Chih-Cheng Lo (羅志成),
Hung-Chang Lai (賴宏昌), and
Wen-Shyen E. Chen (陳文賢)
Current affiliation: Institute of Computer Science National Chung-Hsing University
(國立中興大學資訊科學研究所)
Postal address: No. 250, Guoguang Rd., Nan Chiu, Taichung, Taiwan 402, R.O.C.
(402 台中市南區國光路 250 號)
E-mail address: {loremi, hclai, echen}@cs.nchu.edu.tw
Telephone number: 0928-316788, 04-22840497~901
Fax number: 04-22853869, 04-22622911

(5) Contact author: Chih-Cheng Lo (羅志成)
Telephone number: 0928-316788
loremi@cs.nchu.edu.tw

(6) Keywords: HFC Networks, CATV Networks, MCNS DOCSIS, IEEE 802.14

An Adaptive contention period control in HFC Networks

Chih-Cheng Lo, Hung-Chang Lai, and Wen-Shyen E. Chen
Institute of Computer Science
National Chung-Hsing University
Taichung, Taiwan
E-mail: {loremi, hclai, echen}@cs.nchu.edu.tw

Abstract

As the deployment of the cable TV (CATV) network becomes ubiquitous, the CATV networks have emerged as one of primary technologies to provide broadband access to the home. Due to the natural of CATV network－the radical asymmetric bandwidth, the upstream channel is critical to allocating and scheduling. The objective of this paper is to propose a new adaptive contention period control algorithm in HFC networks to improve the utilization and throughput of the upstream bandwidth. We proposed our method in Section 4 and simulated results in Section 5. Through the simulation, we have shown that in any cases, our adaptive method performs better than MCNS DOCSIS.

1. Introduction

In this new millennium, multimedia and broadband services are provided widely over the Internet. Several emerging wireline and wireless access network technologies to provid broadband access to the home, such as HFC, xDSL, FTTx, and LMDS/MMDS access networks [1]. As ubiquitous deployment of the cable TV (CATV) network, the CATV network has emerged as one of primary technologies to provide broadband access to the home. There are many organizations proposed the MAC layer protocols as the standard of modern HFC networks. In the paper, we propose an adaptive contention slots allocating scheme of HFC networks to improve the throughput and request delay of DOCSIS HFC networks.

A number of organizations have worked hard for many years in order to define open standards for CATV network systems [2-3]. The major associations working on HFC networks are the Multimedia Cable Network System (MCNS) Partners Ltd., the IEEE working group 802.14 [4], the European Cable Communication Association (ECCA), the Digital Audio Video Council (DAVIC) and the Digital Video Broadcasting (DVB). DVB and DAVIC work closely together are tightly connected to the European Telecommunication Standards Institute (ETSI). While both DOCSIS and IEEE 802.14a were developed to facilitate the interoperability between stations and headends designed by different vendors. The MCNS, a consortium consisting of predominantly North American cable operators and media companies, was developed to create a quick and uniform interface specifications for standard, interoperable, for transmission of data over cable networks. Those documents are commonly referred to as the Data Over Cable Service Interface Specification (DOCSIS) [5] and DOCSIS v1.0 was approved as a standard by the ITU on March 19, 1998. Although IEEE 802.14 had been developed into a standard, but the lack of progress motivated these MSOs to develop their own standard in the hopes of accelerating the growth of the market. Due to the delayed progress, the IEEE 802.14 Working Group was dispersed in March 2000, and IEEE 802.14a will remain as a draft thereafter.

Besides above associations, there were other standards associations working on topics related to HFC networks including the Internet Engineering Task Force (IETF) IP over Cable Data Network Working Group [6], the ATM Forum Residential Broadband Working Group [7], the Society of Cable Telecommunications Engineers [8-9], and ITU. However, the implementation of the DOCSIS specification has the dominance of the market.

The main goal of this paper is to propose a new adaptive contention period control algorithm in HFC networks to improve the utilization and throughput of the upstream channels. The remainder of this paper is organized as follows. Section 2 presents the overview of HFC networks, including the network topology and the characteristics of the bandwidth utilizations. In Section 3, the MAC protocol adopted in MCNS DOCSIS and IEEE 802.14a will be discussed. Our research, idea and methods, including modeling analysis and algorithms are presented in Section 4. In Section 5, we present our simulation results in two different aspects. Finally, we make a brief conclusion in Section 6.

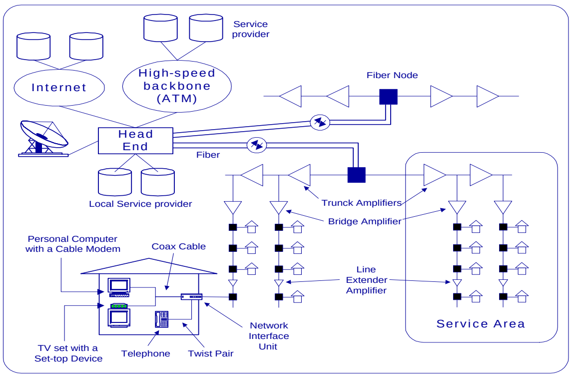


Figure 1. The HFC network architecture.

2. HFC networks overview

In this section we give a brief overview of the HFC networks and discuss the architecture and the main functionality of its PHY and MAC layer protocols. Figure 1 illustrates the architecture of the HFC network. The HFC network possesses the following features that deeply affect the MAC protocol design [2]:

- Point-to-multipoint downstream and multipoint-to-point upstream — since the HFC network uses a hierarchical tree-and-branch topology. It is eventually a point-to-multipoint in the downstream direction, but a multipoint-to-point bus-like access network in upstream direction. Each upstream channel is a multi-access channel, and collisions occur when multiple subscribers transmit simultaneously on the channel.
- The inability to detect collisions by stations —unlike usual bus like networks such as Ethernet, in CATV networks, stations can only listen to the downstream traffic, but not to the upstream traffic. Thus, stations depend on the headend to notify them of the results of upstream transmissions.
- Large propagation delay — since HFC

network is developed for metropolitan transmission topology. The maximum round-trip-delay (RTD) is longer then the LAN topologies. Therefore, a channel utilized to transmit other data frames during the RTD of a transmitted data frame. Furthermore, the MAC protocol should have a ranging scheme to measure the propagation delay for each station to synchronize with the headend.

- Asymmetric upstream and downstream－a high degree of asymmetric on upstream and downstream channel bandwidth. The downstream data rate is substantially much larger then that of the upstream. Thus, the efficiency of upstream channel is crucial. Figure 2 shows the HFC network bandwidth allocation.

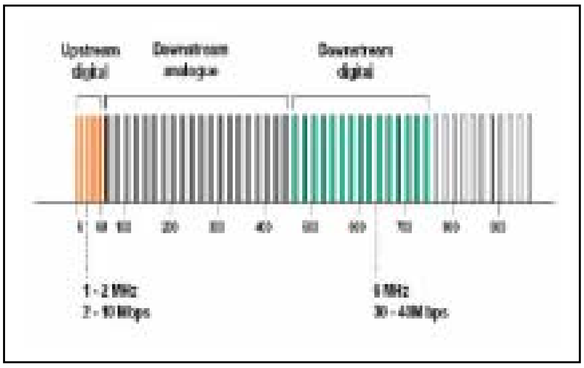


Figure 2. HFC network bandwidth allocation.

- Non-uniform user distribution－usually, the HFC network has as many as 2000 subscribers attached at the leave of the tree. There propagation delay to the headend is quite close to each other. Repeated collisions may occur if the ranging algorithm does not consider this factor.

When a station is powered on, it should be executing the following initialization procedures to startup it's operation [4,5]. There are channel acquisition, obtain upstream parameter, ranging, IP layer establishment, and registration. After initialization is completed, the station enters transmission idle state to wait for the arrival of requests and listens to downstream information.

The PHY layer protocol specifications define the electrical characteristics of the cable, such as the modulation technique, symbol rates and frequencies used [10]. They also defined several operations performed at the headend such as scrambling, Forward Error Correction (FEC), ranging and time synchronization. The MAC protocol's main responsibility is to ensure each station is granted permission to send data to the headend without colliding with other stations that want to send data at the same time and channel, since the upstream channels are shared by all of the stations. If there is any collision, a Collision Resolution Protocol (CRP) is invoked in order to resolve collisions resulting from two or more stations requesting transmitting simultaneously.

Both in MCNS and IEEE 802.14a, the upstream channel is modeled as a stream of minislots, the smallest transmission units. Basically, there are two types of minislots: request slots and data slots; both of them are assigned by the headend. Request slots are subject to carry bandwidth requests made by stations before transmission data, while data slots are used by stations for sending data to the headend. The headend periodically broadcasts a bandwidth allocation MAP, which contains the upstream bandwidth allocation information to notify the stations. Stations learn the assignments from that MAP and work accordingly.

3. MAC protocol: MCNS DOCSIS vs. IEEE 802.14a

Since the development of DOCSIS v1.0 followed closely the IEEE 802.14a specifications, the MAC protocols described in the MCNS and IEEE 802.14a specifications are fundamentally similar: including virtual queue, minislot, downstream MPEG-II format, security module, piggybacking, synchronization procedure, and modulation schemes [2]. However, there are two major differences that may have a direct impact on performance, namely the mapping of higher layer traffic and the upstream contention resolution algorithm [10].

The framing structure of the MCNS is significantly different from the one adopted in the IEEE 802.14a. The MCNS DOCSIS proposed a more suitable IP environment. 6 bytes of MAC header are added to every packet regardless of weather it is an ATM cell or an LLC packet. Nevertheless, since IEEE 802.14 specification intends to provide a complete support of ATM environment and in order to minimize the MAC layer overhead, one byte is added to each ATM cell to form a MAC data PDU as shown in Figure 3, where the ATM layer VPI field is used as part of the 14-bit local station ID. Furthermore, every station must be capable of AAL5 segmentation and reassembly in order to carry IP/LLC traffic.

As for the contention resolution algorithms, DOCSIS adopts Binary Exponential Backoff algorithm to resolve collisions in the request minislot contention process. However, it allows flexibility in selecting the Data backoff start (DBS) and Data backoff end (DBE) window sizes to indicate the initial and maximum backoff window size used in the algorithm. The headend controls the initial access to the contention slot by setting Data backoff start and Date backoff end. When a station has data to send, it sets its internal backoff windows according to the data backoff range indicate in the allocation MAP. The random value means the number of contention transmit opportunities, which the station must defer before transmitting. The station then randomly selects a number within the backoff range and sends out its request. Since station cannot detect whether it is collision or not, it should wait the headend sends feedback either a Data Grant or an Acknowledgement (Ack) in a subsequent allocation MAP. If station does not receive either Data Grant or Ack in the subsequent allocation MAP, it means a collision has occurred. However the headend does not need to send an explicit feedback message on the status of each contention slot as in the IEEE 802.14a specifications. In this case, the station must then increase its backoff windows by a factor of two as long as it is less than the Data backoff end value set in the allocation MAP. Once again, the

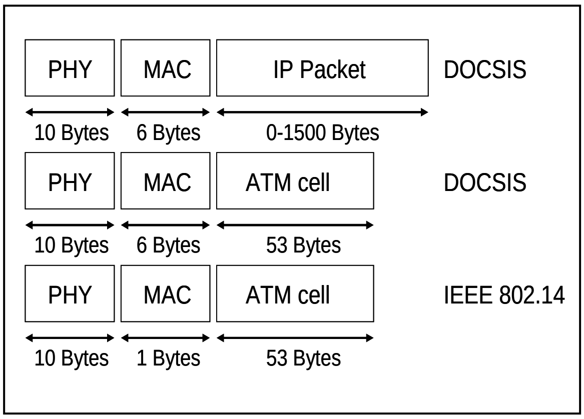


Figure 3. DOCSIS and IEEE 802.14a mapping of higher layer

station randomly selects a number within its new window range and repeats the contention process depicted above. After 16 unsuccessful retries, the station discards the MAC PDU.

The collision resolution algorithm in the IEEE 802.14a consists of two parts. The first part is the first transmission rules that adopt priority plus FIFO algorithm designed for newcomers, while the second part is the retransmission rule that adopts *n*-ary tree plus p-persistent algorithm designed for collide requests. The headend controls the initial access to the contention slots and manages the Contention Resolution Protocol (CRP) by assigning a Request Queue (RQ) number to each contention slot. When a data packet is received, the station generates a Request Minislot Data Unit (RMDU). The headend controls the stations entry by sending an Admission Time Boundary (ATB) periodically. Only stations that generated RMDU time before ATB are eligible to enter the contention process. Once the RMDU is generated, the station waits for a CS allocation message from the headend that reserves a group of CS with RQ = 0 for newcomer transmission. As upstream channel is shared media, multiple stations may attempt to send their requests simultaneously and a collision may occur. A feedback message is sent by the headend to the station after a roundtrip time to inform the status of the contention slots. With a successful request transmission, a Data Grant message will be sent by the headend and the station activates its data transmission state machine. In case of collision, the feedback message contains a particular RQ value to inform those stations involved. The stations need to retransmit their requests within the specific contention slot group that contains the equal RQ value. The contention slot groups are usually allocated in the order of decreasing RQ values. For each RQ value the headend assigns a group of contention slots and associated splitting value (SPL) that is default to three. i.e., IEEE 802.14a adopts 3-ary tree algorithm to process contention.

4. Adaptive contention period control

In this section, we propose an adaptive contention period control scheme to predict the number of contention slots in order to better cope with the request contention and to provide a better system performance. Some papers have studied this topic [11-15]. The idea of our research is how to dynamically adjust the number of contention slots to meet the number of requests arrived at the system, and to achieve the maximum bandwidth throughput and minimum request delay time. We first analyze the probability of contention, and then derive the bandwidth utilization and predict the reserved number of contention minislots. Finally, we propose a new adaptive method to dynamically adjust the number of contention slots to realize our ideas.

4.1 Investigation probability of contention

For each contention minislot in the upstream direction, there have only three states, *Idle*, *Success*, or *Collision*. It is intuitive that we should expect more Success slots but less Idle and Collision slots, since the latter two cases will waste the bandwidth. The following shows how we derive the probability of those three cases. Suppose we have *m* minislots and *n* requests, where *m* and *n* denote the number of contention slots and the number of requests, respectively. Then we can derive the total combinations of the permutation as:

$$\binom{n+m-1}{n} \qquad (1)$$

And the combinations of the permutation if there is only one empty minislot and exactly one monislot has been selected by one request were shown in equations (2) and (3), respectively.

$$\binom{n+(m-1)-1}{n} \qquad (2)$$

$$\binom{(n-1)+(m-1)-1}{n-1} \qquad (3)$$

Based on the aforementioned formulas, we can derive the probability of *Idle slots*, *Success slots*, and *Collision slot*s as shown in equations (4), (5), and (6), respectively.

$$\frac{(2)}{(1)}=\frac{\binom{n+m-2}{n}}{\binom{n+m-1}{n}}=\frac{\left(\frac{(n+m-2)!}{n!(m-2)!}\right)}{\left(\frac{(n+m-1)!}{n!(m-1)!}\right)}=\frac{m-1}{n+m-1},$$

(4)

$$\frac{(3)}{(1)}=\frac{\binom{n+m-3}{n-1}}{\binom{n+m-1}{n}}=\frac{\left(\frac{(n+m-3)!}{(n-1)!(m-2)!}\right)}{\left(\frac{(n+m-1)!}{n!(m-1)!}\right)}=\frac{n(m-1)}{(n+m-1)(n+m-2)},$$

(5)

and

$1-(4)-(5)=$

$$1-\frac{m-1}{n+m-1}-\frac{n(m-1)}{(n+m-1)(n+m-2)}$$
$$=\frac{n(n-1)}{(n+m-1)(n+m-2)}$$

(6)

The objective of our proposal is to reserve the appropriate number of contention slots that meet the maximum success probability when *n* number of requests had arrived, such that, we can take differentiation the equation (5) with respect to *m*, and find out the maximum value of *m* as shown in equation (7). From (7), we obtain that *m* approximates *n*, i.e., if there are number of *n* requests arrived, the system should reserve *m* contention slots to achieve the maximum success probability.

$$m=\sqrt{n(n-1)}+1 \qquad (7)$$

4.2 Dynamic adjustment contention slots method

After analyzing the value of maximized success probability, we will discuss how to dynamically adjust the contention slots based on the previous probability of idle/success/collision. Since our idea is to make the appropriate number of the contention slots predictable, we first define the following notations to depict our algorithms:

D_{map}: time between the CMTS starts transmitting a MAP and when the MAP goes into effect

N_p: the request had been received by CMTS successfully but have not allocate data slot yet

N_{cm}: total number of CMs in this HFC networks

P_{idle}: the probability of slot as idle state in the previously contention slot

$P_{success}$: the probability of slot as success state in the previously contention slot

$P_{collision}$: the probability of slot as collision state in the previously contention slot

T_{idle}: the threshold of idle state's probability

$T_{success}$: the threshold of success state's probability

$T_{collision}$: the threshold of collision state's probability

N_{pc}: the number of contention slots in previously competition

N_c: the number of contention slots in this cycle's competition

N_{max}: the maximum number of contention slots

N_{min}: the minimum number of contention slots

Weight: the weighting value

We can derive from above discussion that the range of contention slots number must be between the follwing two numbers:

$N_{max} = N_{cm} - N_p$

$N_{min} = D_{map}$

Since our prediction concept is to observe the previous probability of success/collision and whether that reserved minislots are suitable or not as the basis. Nevertheless, not only previous basis, we also have to include a set of thresholds as the adjusting foundation. The set of values (idle, success, collision) in our simulation is (45%, 40%, 35%), respectively. While the object of weighting function is to accelerate the adjustment of the number of collision slots when the probability of collision is greater than the successes. Based on these concepts, we can derive our algorithms for different conditions as follows:

```
if ($P_{collision} > P_{success}$)
    weight++;
else
    weight = 0;
```

```
If ( $P_{collision} > T_{collision}$ )
{
    if ( weight > 2)
        $N_c = N_{cm} - N_p$; // set to maximum value
    else
        $N_c = N_{pc} + \max(D_{map}, N_{pc})$*weight;
};
```

```
If ( $P_{idle} > T_{idle}$ )
    $N_c = N_{pc} - (N_{pc} * P_{idle} - N_{pc} * P_{success})$ ;
```

```
If ( $P_{success} > T_{succes}$ )
    $N_c = N_{pc}$ ;
```

5. Simulation result

In this Section, we will compare our method with DOCSIS specification and point out the difference. For practicality, we measure the throughput, request delay, and drop numbers of the simulated system. Where the throughput was defined as how much data (in Mbps) can be transmitted in a unit time period. The request delay is the time it receives a transmission acknowledgement by the headend from the time the request arrives at the station. We assume that data arrives at the MAC layer of a station is small enough to fit into a single data slot. The request delay includes the waiting time for a newcomer slot, delays due to collision resolution, scheduling delay of a grant message at the headend, and transmission delay of the data slot. The drop number are the amount of packets dropped by the queue of a cable modem. In general, the condition of packet dropped is caused by the buffers of CM is full or that retry number due to collision is greater then the upper bound, i.e., 16.

Parameter	**Value**
Upstream channel capacity (QPSK)	2.56 Mbps
Downstream channel capacity	26.97 Mbps
Minislot	16 bytes/minislot, 50 usec/minislot
Number of contention slots in a MAP (*)	40 minislot ➔40 * 16 = 640 bytes ➔40 * 50 = 2 msec
MAP size	50 minislot (100%) ~ 2048 minislot(2%) ➔800 ~ 32768 bytes ➔2.5 ~ 102.4 msec
Maximum number of IEs in a MAP	240
Number of CMs	50~600
One way delay	0.5 msec
DMAP time	2 msec
Data size	64 bytes
Backoff limit (DOCSIS only)	6 ~ 10
Maximum retry	16
Number of queue for data in CMs	30

Table 1. Simulation Parameters

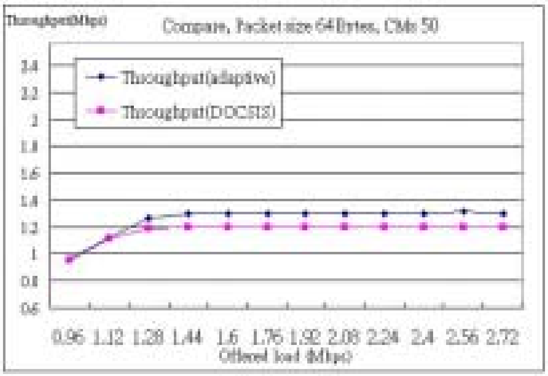
*: We assume the number of contention slots in DOCSIS is fixed at 40 slots.

As most of the subscribers attached at the leaves of the HFC networks, we assume that all of them have the same distance with the headend, and all have the Poisson arrival rate λ, and with the simulation parameters listed in Table 1.

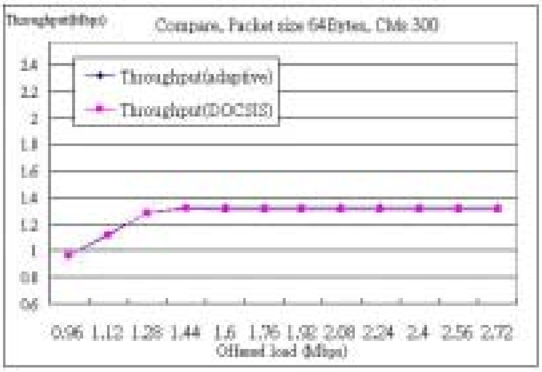
The following simulation studies separate the performance of the proposed mechanism in two aspects. The first one fixes the number of CMs, but varies with the offered load, throughput, and request delay. The other group fixes the offered load, but varies with the number of CMs, throughput, request delay, and drop number. Figures 4 (a) to (c) show the results of the simulation for the first aspect discussed above.

As we observed in Figure 4, when the offered load is less than 1.28 Mbps, there is no difference between our method and DOCSIS. But when the offered load is greater then 1.28 Mbps, the difference becomes significant gradually. Another checkpoint is when the offered load reaches 1.44 Mbps, the throughput does not change after that value.

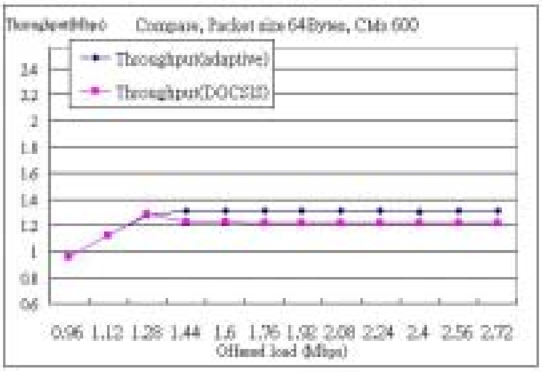
As a result, that point can be viewed as the watershed to saturation state. Before that point, our adaptive scheme obtains a better throughput



(a)



(b)

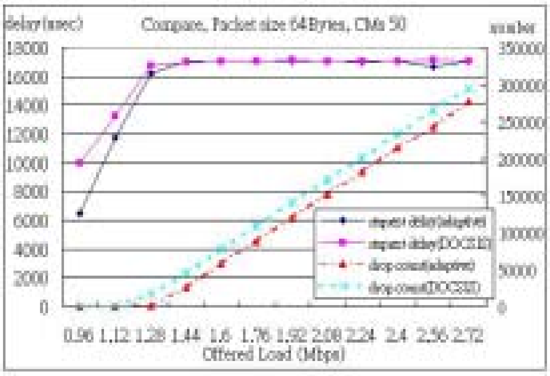


(c)

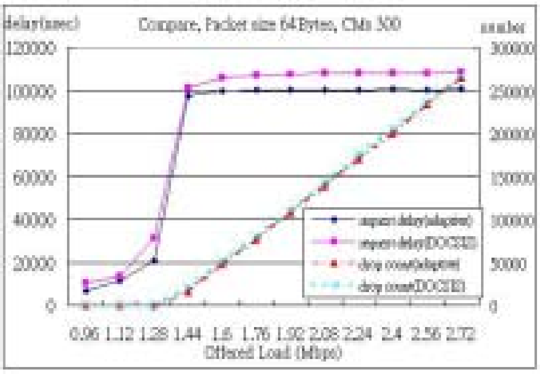
Figure 4. The relationship between throughput and offered load.
(a) CMs = 50 (b) CMs = 300 (c) CMs = 600

than DOCSIS for about 7%.

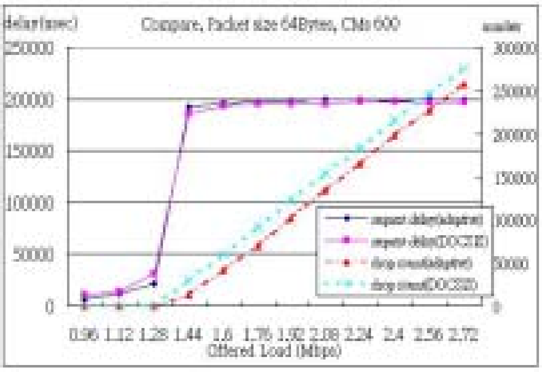
The other comparisons are the relationship between request delay, drop number and offered load. As shown in Figure 5, the number of CMs is fixed at 50, 300, and 600 respectively, but request delay, offered load, and drop number are variable.



(a)



(b)



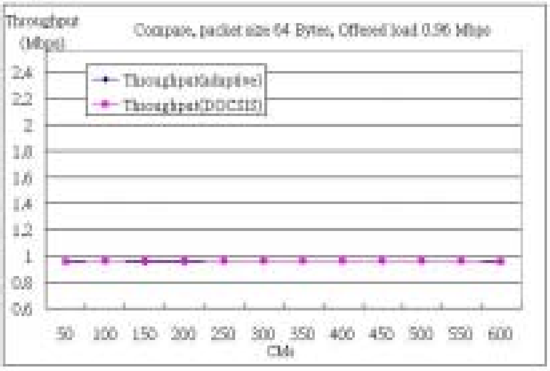
(c)

Figure 5. The relationship between request delay, drop number and offered load.
(a) CMs = 50 (b) CMs = 300 (c) CMs = 600

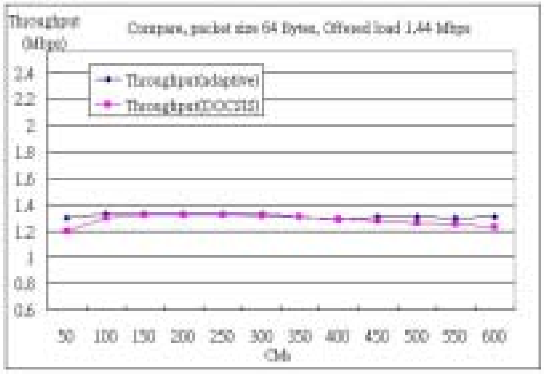
As shown in Figure 5, the solid line represents the request delay, while the dotted line represents the drop packet numbers. In the request delay item, the curve tendency of our method is similar with DOCSIS. When offered load below then 1.28 Mbps, the increase is smoothly. While if offered load over that point, the request delay will take off sharply until offered load equal 1.44 Mbps, and the increasing will about 9 times of original value. After 1.44 Mbps, the curve retain horizontally. Let’s observe the other parameter, the drop number almost stays zero when offered load is less then 1.28 Mbps, but it increases obviously when offered load is greater than that point. There is an interesting phenomenon, i.e., when the difference between our method and DOCSIS in request delay is small, then the difference in drop number will get larger. But, when the difference between our method and DOCSIS in request delay getting larger, then the difference in drop number gets smaller.

We discuss another issue of our simulation now, to fix the offered load. We depict the simulation result in Figure 6. As we have seen in these figures, when offered load is less then 1.44 Mbps, there are almost no difference between our methods and DOCSIS. But when offered load getting greater than 1.44 Mbps, there is about 7% difference between them.

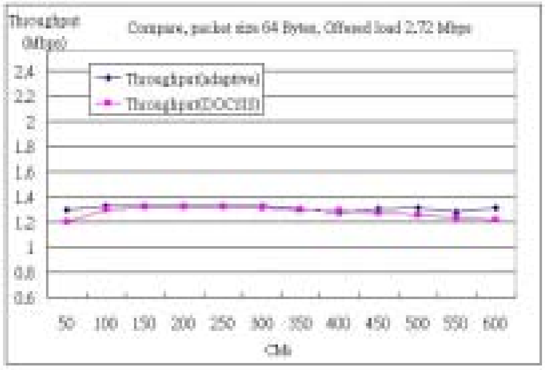
In our last experiment, we observed the relationship between request delay, drop number, and the number of CMs while fixed the offered load. Again, the solid line represents the request delay, while the dotted line represents the drop packet numbers.



(a)



(b)

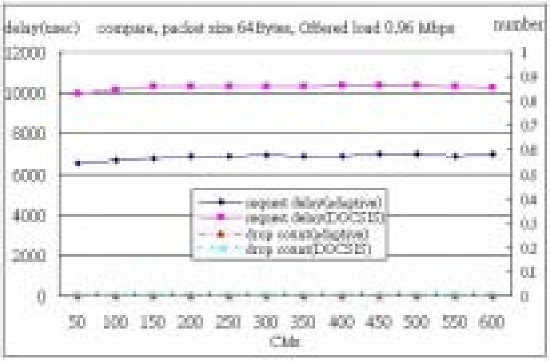


(c)

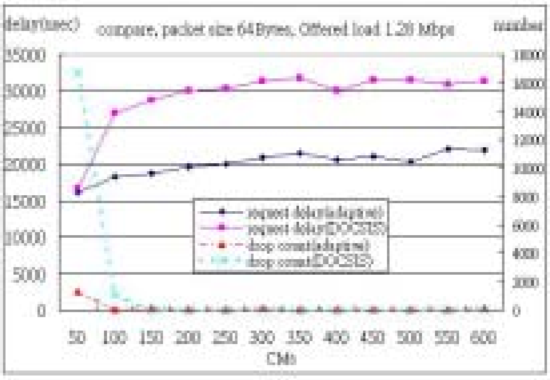
Figure 6. The relationship between throughput and offered load

(a) offered load = 0.96 (b) offered load = 1.44 (c) offered load = 2.72

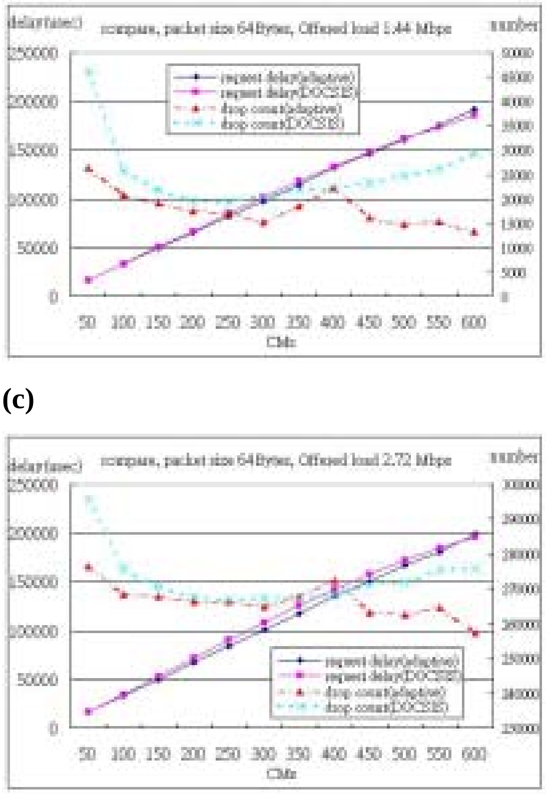
As shown in Figure 7, when offered load is



(a)



(b)



(c)

(d)

Figure 7. The relationship between request delay, drop number and offered load.

(a) offered load = 0.96 (b) offered load = 1.28 (c) offered load = 1.44 (d) offered load = 2.72

less than 1.28 Mbps, in the item request delay, our method is about 20% to 50% better than DOCSIS. But the difference diminishes when offered load is greater than 1.44 Mbps. Nevertheless, when offered load is light, the drop numbers of both methods are almost zero. Whereas, when the offered load gradually increases and exceeds 1.44 Mbps, the drop packet number increases dramatically, especially in DOCSIS case.

6. Conclusion

In this paper, we have presented the architecture of the HFC networks and pointed out that the upstream channel is one of the major factors affecting the network performance and the channel access allocation mechanism is very important in improving the overall throughput. We propose an adaptive method to predict the suitable number of contention slots to meet the request numbers arriving at the system; i.e., if there are *n* arriving requests, then the system should reserve *m* contention slots to achieve the maximum success probability. We also evaluate the performance of the system we proposed in two different aspects by experiments. From the simulation result, we conclude that our adaptive method has a better performance than that of MCNS DOCSIS in the case studied.

[REFERENCES]

[1] Gorge Abe, “Residential Broadband, Second Edition,” Cisco Press, 2000.

[2] Y. D. Lin, W. M. Yin, and C. Y Huang, “An Investigation into HFC MAC Protocols: Mechanisms, Implementation, and Research Issues,” *IEEE Communications Surveys*, PP. 2-13, Third Quarter 2000.

[3] V. Sdralia, C. Smythe, P. Tzerefos, and S. Cvetkovic, “Performance Characterisation of the MCNS DOCSIS 1.0 CATV Protocol with Prioritised First Come First Served Scheduling,” *IEEE Trans. Broadcasting*, vol. 45, no. 2, pp. 196-205, Jun. 1999.

[4] Institute of Electrical And Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017, USA, IEEE project 802.14/a Draft 3 Revision 1, Apr. 1998.

[5] Cable Television Laboratories, Inc., Data-Over-Cable Service Interface Specifica- tions-Radio Frequency Interface

Specification, Version 2.0, Nov. 2001.

[6] M. Ahed, and G. Roeck, "IP Over Cable Data Network (IPCDN) Service," IETF Draft, 1996.

[7] Residential Broadband Architecture Framework, ATM Forum Technical Committee, AF-RBB-0099.000, Jul 1998.

[8] ITU-T, Recommendation J.83 Series J: Transmission of Television, Sound Programme and Other Multimedia Signals, ITU-T, April 1997.

[9] ITU-T, Recommendation J.112: Transmission Systems for Interactive Cable Television Services, ITU-T Pre-published, March 1999.

[10] N. Glomie, F. Mouveaux, and D. Su, "A Comparison of MAC Protocols for Hybrid Fiber/Coax Networks: IEEE 802.14 vs. MCNS," *Proc. ICC 1999*, pp. 266-272.

[11] D. Sala, J. O. Limb, "A Protocol for Efficient Transfer of Data over Fiber/Cable Systems," IEEE Trans. On Networking, vol. 5, no. 6, Dec. 1997

[12] N. Glomie, Y. Saintillan, and D. Su, "A Review of Contention Resolution Algorithms for IEEE 802.14 Networks," *IEEE Communications Surveys*, First Quarter 1999, PP. 2-12.

[13] Y. D. Lin, C. Y Huang, and W. M. Yin, "Allocation and Scheduling Algorithms for IEEE 802.14 and MCNS in Hybrid Fiber Coaxial Networks," *IEEE Trans. on Broadcasting*, vol. 44, pp. 427-435, Dec. 1998.

[14] W. M. Yin, and Y. D. Lin, "Statistically Optimized Minislot Allocation for Initial and Collision Resolution in Hybrid Fiber Coaxial Networks," *IEEE Journal on Select Area in Commun.*, vol. 18, no. 19, pp. 1764-1773, Sep. 2000.

[15] M. D. Corner, J. Libeherr, 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. On Networking*, vol. 8, no. 2, pp. 200-211, Apr. 2000