

## 摘 要

当今社会已经进入了信息社会,对信息的获取和传送方式提出了极高的要求。传统的电话接入网方式已经很难再继续满足日益增长的带宽需求了,HFC 宽带接入网正在成为一大主流。它具有覆盖面广和带宽资源丰富的特点。DOCSIS 协议在 HFC 网络中应用广泛,使其已成为事实的标准。由于 HFC 宽带接入网的结构特点,上行信道是多点对单点的通信,并且采用了时分复用和带宽请求/授权机制,所以存在一系列的技术难点。上行信道长期以来一直备受重视,是 HFC 宽带接入网络技术中的研究热点。

本文对基于 DOCSIS 的 HFC 宽带接入网上行信道的几个关键技术做了详细的研究,主要分成三个部分:

(1) 分析了上行信道存在的主要问题,提出了对 HFC 网络的改进,从而使 CMTS 具有了检测冲突的能力。设计了基于冲突检测的竞争解决算法,可以根据冲突的剧烈程度,调整回退窗口的大小,有效地减少冲突,以适应用户数目的不断变化。

(2) 研究了冲突解决和接入延时之间的矛盾问题。设计了基于优先级的冲突解决算法,成功解决了这一矛盾。本算法使高优先级用户获得了低延时的接入。按照优先级的高低,既保障了高优先级用户的冲突解决和首次接入,又尽可能地服务于低优先级用户。

(3) 分析了竞争时隙和数据时隙之间存在的互动关系。根据这个关系,设计了动态的时隙分配算法,验证了该关系在网络运作中的实际作用。

**关键词: HFC DOCSIS 冲突解决 时隙分配**

## Abstract

Nowadays, the society has become "Information Society". People demand much more effective manners to gain and transmit information. The traditional PSTN access network cannot satisfy the urgent need any more. Thus, HFC broadband access network is now coming on the "stage" with the specialty-widely covering and abundance bandwidth. The DOCSIS specification is established as the primary HFC network data communication standard. Because of the structure character of HFC, DOCSIS makes use of upstream channel in TDMA mode and allocate it by "request-grant" mechanism. So there are a series of technical difficulty. The technology of upstream channel access is a key technology, and it is always the focus of the research.

The key technologies of upstream channel of the HFC are under research, base on DOCSIS. This paper is concentrated on the following three aspects:

(1)The primary problems of upstream channel are analyzed, and the mend of HFC is present. It makes CMTS to be provided with the ability to detect the collision. The collision resolution algorithm based on detecting collision is designed. It can adjust the "backoff windows" according to the degree of collision. Thus, it can decrease the collision effectively to adapt the change of user.

(2)The contradiction of collision resolution and access delay is researched. The collision resolution algorithm based on priority is designed to solve the contradiction. It makes the high priority user to have lower access delay. According to the priority, the collision resolution and the fist-time access of the high priority user are guaranteed, and the low priority user is served by its best effort.

(3)The connection of contention slot and data slot is analyzed. According to the connection, the dynamic allocation of slots is designed. It validates the effect of the connection in the networks.

**Key words: HFC, DOCSIS, collision resolution, slot allocation**

# 独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的资料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

签名: 马俊康 日期: 2005年5月21日

## 关于论文使用授权的说明

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行查阅，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密的学位论文在解密后应遵守此规定)

签名: 马俊康 导师签名: 马俊康  
日期: 2005年5月

## 缩略语

CBR	Constant Bit Rate	恒定数据业务
CM	Cable Modem	电缆调制解调器
CMTS	Cable Modem Termination System	电缆调制解调器终端系统
CSMA	Carrier Sense Multiple Access	载波监听多路访问
CSMA/CD	Carrier Sense Multiple Access with Collision Detection	载波侦听多路访问/冲突检测
DOCSIS	Data Over Cable System Interface Specification	电缆数据业务接口规范
HFC	Hybrid Fiber Coaxial	光纤同轴混合网
MAC	Media Access Control	媒体接入控制
MCNS	Multimedia Cable Networks System	多媒体电缆网络系统
PSTN	Public Switched Telephone Network	公共电话交换网
VOD	Video on demand	视频点播
VoIP	Voice Over IP	语音 IP

## 第1章 引言

### 1.1. 概述

当今社会已经进入了信息社会, 信息对于人们的生活和工作起到了至关重要的作用。大量信息极大地改变了工作和生活的方式, 同时, 人们对信息的依赖和需求又不断增加。因此, 这就对信息的获取和传送方式提出了极高的要求。近十多年来, 网络的高速发展正是这一需求的最直接体现。正如半导体行业的摩尔定律(即微处理器的速度每 18 个月增长一倍)揭示了芯片的发展, 网络上的业务量也在以极快的速度成倍增长。据不完全统计, 网络业务量每 6 个月就会增加一倍。近年来, 各种多媒体服务出现并迅速增长, 如视频会议, 视频点播(VOD: Video on demand), VoIP (Voice over IP), 网络游戏等等。增长的网络业务量对网络带宽有了更高的要求, 这就对传统的接入网络提出了挑战。

传统的接入网络使用的是电话网络, 即 PSTN<sup>[1]</sup> (Public Switched Telephone Network)。通过不同的调制方法, 可以得到不同的带宽。过去使用 ISDN, 其带宽只能到达 128Kbps。即便是现在广泛应用的 ADSL, 其理论上行带宽也只有 2Mbps, 下行 8Mbps。传统的电话接入网方式已经很难再继续满足日益增长的带宽需求了, 因此其它的宽带接入网方式也就应运而生。其中, HFC 宽带接入网正在成为一大主流。

HFC (Hybrid Fiber Coaxial) 即混合光纤同轴电缆, 而建立在此基础上的即是 CATV (Community Antenna Television) 网络。HFC 宽带接入网有其特有的优势: 首先, 带宽资源非常丰富, 下行方向的带宽可达几百 MHz, 数据传输速率为 2~40Mbps。其次, 有线电视网在世界上许多国家覆盖面非常广, 已经建设完善, 易于控制潜在数据业务客户的接入成本。我国有世界上最大的有线电视网络和最多的有线电视用户, 所以在我国发展 HFC 宽带接入网络前景广阔。

因此, 研究 HFC 宽带接入网有着重要的意义。现在世界上已经有多个成熟的 HFC 宽带接入网标准, 我们选用了其中使用最为广泛的 DOCSIS (Data Over Cable Service Interface Specification) 标准。

## 1.2. 主要工作

本文主要研究了基于 DOCSIS 标准的 HFC 宽带接入网上行信道的几个关键技术。由于 HFC 宽带接入网的结构特点, 上行信道是多点对单点的通信, 并且采用了时分复用和带宽请求/授权机制, 所以存在一系列的技术难点。因此, 如何合理有效地安排网络运作一直是研究者的热点问题。

本文的研究内容主要分成了三个部分: (1) 研究了上行信道的冲突解决, 对 HFC 网络提出了改进, 进而设计了新的竞争解决算法。(2) 研究了上行信道的时隙分配和延时特性的关系, 设计了新的时隙分配和冲突解决算法。(3) 研究了时隙间存在的互动关系, 分析和设计了在这种关系下的时隙分配。

## 1.3. 本文的组织

本文的内容安排如下:

第二章对 HFC 宽带接入网的发展和架构做了一个简要介绍。对现有成熟的网络标准做了介绍, 并比较了标准间的异同点。

第三章分析了 DOCSIS 标准的基本架构, 以及物理层和 MAC 层的定义。重点研究了上行信道的带宽分配和冲突解决是如何运作的。

第四章分析了上行信道存在的主要问题, 提出了对 HFC 网络的改进, 实现了对上行信道的冲突检测, 然后设计了基于冲突检测的竞争解决算法。

第五章研究了冲突解决和接入延时之间的矛盾问题, 设计了基于优先级的冲突解决算法。

第六章分析了竞争时隙和数据时隙之间存在的互动关系, 研究了基于这种动态过程的时隙分配。

第七章总结了全文的工作, 论述了以上三个研究部分之间的联系。

## 第2章 HFC 宽带接入网及其标准简介

### 2.1. 基于 HFC 的 CATV 宽带接入网的发展

#### 2.1.1. CATV 的发展

CATV发展比较早,最早于1950年在美国出现了小区共同天线(Community Antenna Television)<sup>[1]</sup>。后来这种系统逐渐发展到服务商接收电视讯号、或者自行制作节目,再配送到用户家中,此种方式能提供更丰富的频道节目,称为有线电视(Cable TV 或 CATV)。

到70年代,有线电视开始在美国大中城市出现。据调查在1996年时,约有8960万户的家庭连接有线电视网络,普及率约在63%以上。在美国大幅开放电信业务的政策下,一些有线电视公司纷纷跨入电信的领域。只是目前在美国有线电视公司的规模和电话公司相去甚远,造成有线电视公司在新技术的研发能力和电话公司相比起来就显得吃力了些。但是由于有线电视网络先天具有高频宽及高普及率的特性,使得许多有线电视服务商还是积极投入于交互式多媒体应用的研究,所以未来有线电视在美国还是颇有发展潜力的。

在欧洲,荷兰、比利时、卢森堡等国有线电视网络普及率高达90%以上。英国在欧洲国家中,其对有线电视网络的政策开放程度是首屈一指的,从1983年起就出现了第一个双向系统的有线电视网络,之后也开放有线电视得以经营电信业务,其自由化程度为世界之冠。也因此英国是世界上第一个使用有线电视网络传送电话的国家,在1995年7月英国有线电视网络上的电话用户已高达一百万户,因此有线电视网络在英国发展是一片看好。

中国有线电视的发展,走的是一条由局部到整体的路线。各地有线电视的发展一般都是由最初的居民楼闭路电视,发展到小区有线电视互联,进而整个城域(行政区划)有线电视的互联。<sup>[2]</sup>自1990年以后,中国有线电视从各自独立的、分散的小网络,向以部、省、地市(县)为中心的部级干线、省级干线和城域网发展,并成为全球第一大有线电视网。到1998年底,全国有线电视网络线路长度超过240多万公里,其中光缆网超过30万公里。有线电视用户数达8000多万,并仍以每年500万户以上的速度增长。有线电视用户数已稳居世界第一位。有线电视成为我国家庭入户率最高的信息工具。

### 2.1.2. HFC 宽带接入网的发展

进入90年代以后,广播电视全面数字化和因特网的迅速发展给有线电视形成有力的挑战,同时也给有线电视的发展带来了千载难逢的机遇。在广播电视数字化方面,除美国走在了发展的前列以外,德国、西班牙、澳大利亚、英国、日本甚至台湾也制定出了数字化发展计划。充分挖掘其双向传输能力,真正实现宽带高速接入,使有线电视网络在21世纪的信息时代更紧密地与用户连接在一起。

针对机遇与挑战,各有线电视公司采取了一些促进技术及业务发展的措施。首先是对基础网络改造,许多国家花巨资进行光纤传输网络和现代HFC网的改造,使得有线电视的光节点尽量靠近家庭,例如每一个光节点仅带不到200个家庭。其次是在业务开展方面,通过数字技术的引入,为用户提供上百套的数字电视节目和新型的宽带数据业务。目前在美国,有线电视公司纷纷采用电缆调制解调器(即Cable Modem,后文将详细讲解)。实践证明,Cable Modem相对于电信公司的ADSL有更先进的优越性,所以计算机公司如微软,电信公司如AT&T纷纷把目光看向了有线电视。我国新建网CATV网络大部分为HFC网,广电总局要求新建HFC网每个光节点用户在2000户以下,有利于数据传输和宽带接入。目前,我国正在建更大规模的HFC宽带接入试验网。

现在,HFC已经成为有线电视系统的主要模式。随着社会信息化的进程,有线电视网、电信网和计算机网“三网融合”在发达国家已经开始,HFC网络的发展进入一个全新的时期。

## 2.2. HFC 网络架构

最初的CATV网络使用的是同轴电缆。但是随着技术和对网络服务的要求的提高,同轴电缆早已不能满足需要了。于是CATV网络已经从单向的模拟视频信道广播转变为了光纤/同轴混合网(HFC),并且正在从单向发展到双向,同时能传送模拟/数字视频信道和高速数据。HFC网络可以分成三个部分<sup>[3]</sup>:头端(Headend)、传输线路、用户端,如图2.1所示。

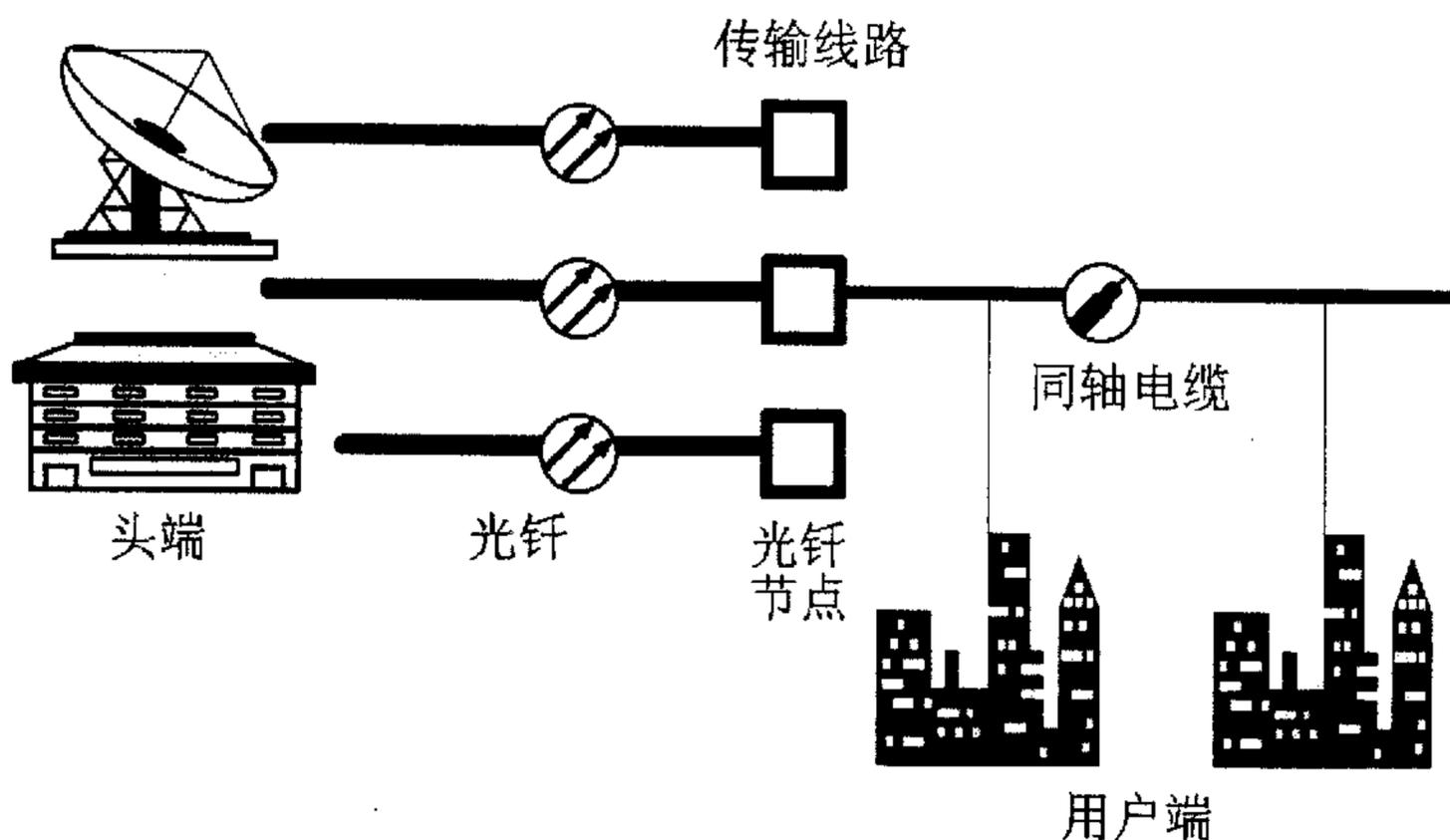


图 2.1 HFC 网络架构

头端(Headend)是负责控制及维护整个有线电视网络的运作。传统有线电视网络的头端,单纯只负责播放节目或接收卫星节目讯号,并把节目内容经传输线路,传送到客户端。但是现今有线电视网络均被要求能传送数字的双向信号,头端要负责协调双向数据的传送顺序,因此头端的工作加重了许多。

传输线路是负责传送头端及客户端两者间信号,依照其布线位置还可分为干线、馈线等多个部分。目前 HFC 网主干部分采用光纤,光节点到配线盒使用同轴电缆,配线盒到用户端使用树枝状结构分配型同轴引入线。

客户端是指使用 HFC 网的用户家中。在传统的 CATV 网中,用户使用选台器或电视机和 HFC 网连接,但在数字传送的 HFC 网中,用户则是使用电缆调制解调器(Cable Modem)或机顶盒(即有线电视译码器)连接 HFC 网络。

### 2.3. HFC 网络标准

HFC 网络是一个树型的拓扑网络,下行信道只能由头端使用来给用户端传送数据,所以下行方向的传送相对容易。但是上行信道是共享信道,多个用户端对一个头端,即多点对单点的传输,所以当多个站点同时进行上行传输时就会发生冲突,造成传输错误。因此,为了能在 HFC 网络上成功有效地传送数据,必须要有一个针对 HFC 网络特性所设计的标准。世界上有多个组织都制定了各自的标准,其中一些已经成为了较为通用的行业标准,下面介绍其中影响最大

的几个标准。

### 2.3.1. IEEE 802.14 标准

1994 年 5 月，国际电子、电机工程师协会<sup>[4]</sup>(IEEE) 802.14<sup>[6]</sup>工作小组成立，这是最早的标准制定组织。该标准主要包括有线电视轴缆网络的媒体控制层 (MAC Layer: Media Access Control Layer) 和物理层 (Physical Layer) 的制定。其主要特点是：第一次传送规则为带有优先级的先进先出 (FIFO) 机制，而重传规则是 n 元树算法 (N-ary tree algorithm)。

### 2.3.2. DOCSIS 标准

由于 IEEE 802.14 进展缓慢，美国的有线电视商于 1995 年成立多媒体有线电视网络系统 (MCNS)<sup>[5]</sup>组织。1995 年 12 月，MCNS 组织先后发表了 MCNS 的八个文件，统称为“电缆数据射频接口规范”，即后来的 DOCSIS<sup>[5]</sup> (Data Over Cable Service Interface Specification) 标准。该标准获得全美百分之八十的 Cable Modem 制造商的支持，使得此标准成为 HFC 网络数据通讯标准的主力。在 1998 年 3 月，国际电信联盟接受 DOCSIS 标准，作为 Cable Modem 的标准。1999 年美国 CBF (Cable Broadband Forum) 根据 DOCSIS 标准决定各家调制解调器产品的兼容性，并交由 MCNS 所成立的 CableLabs 给予认证。到目前为止，DOCSIS 已被广泛的使用在世界大部份的地区。其最新的版本为 DOCSIS 2.0。

### 2.3.3. DVB/DAVIC 标准

欧洲的标准体系组织包括 DVB<sup>[7]</sup> (数字视频广播)、DAVIC<sup>[7]</sup> (数字音视频理事会) 以及 ECCA (欧洲有线电视运营商联盟) 三个组织。DVB 和 DAVIC 组织长期致力于数字视频标准的制订。DVB/DAVIC 是 DOCSIS 标准在欧洲的强有力的挑战者。Euro Cable Labs (欧洲有线电视实验室) 在欧洲有线电视联盟的指导下，一直支持以 DVB/DAVIC 的为基础的“欧洲调制解调器”1999 年 5 月欧洲有线电视实验室发布了基于 DVB/DAVIC 的 Euro Modem 规范，DVB Cable Modem 是欧洲经营者的首选。

### 2.3.4. 标准间的比较

上述标准体系的频带划分、带宽及信道参数等方面的规定，都存在较大差异，因而互不兼容。北美标准是基于 IP 的数据传输系统，侧重于对系统接口的

规范, 具有灵活的高速数据传输优势; 欧洲标准是基于 ATM 的数据传输系统, 侧重于 DVB 交互信道的规范, 具有实时视频传输优势。它们之间的比较详见表 2.1<sup>[8][9]</sup>。

表 2.1 DOCSIS, IEEE802.14 和 DAVIC/DVB 比较

	DOCSIS	IEEE 802.14	DAVIC/DVB
上行调制方式	16QAM/64QAM	QPSK/16QAM	QPSK
下行调制方式	64QAM/256QAM	64QAM/256QAM	QPSK
上行信道接入方式	FDMA/TDMA/CDMA	FDMA/TDMA	FDMA/TDMA
QoS	UGS/rtPS/UGS-AD/CIR	CRB/VBR/ABR	---
安全	RSA/DES	RSA/DES	---
冲突解决	二进制指数后退	N 元树算法	二进制指数后退
分组格式	IP 分组	ATM 信元	ATM 信元
上行速率	30Mbps	10Mbps	10Mbps
下行速率	40Mbps	30Mbps	10Mbps

## 2.4. 小结

本章对HFC宽带接入网的发展和架构做了一个简要介绍, 指出了在我国发展HFC宽带接入网的重要意义。对现有成熟的网络标准做了介绍, 并比较了标准间的异同点。

## 第3章 DOCSIS 协议分析和研究

### 3.1. DOCSIS 的基本架构

DOCSIS制订的主要目的是在于让客户端的计算机的数据可以经过CM (Cable Modem: 电缆调制解调器) 连接到HFC有线电视网络上, 并经由头端和外界网络做双向的沟通。图3.1所示即说明了这种架构, 图中CMTS (Cable Modem Termination System: 电缆调制解调器终端系统) 即是指头端。

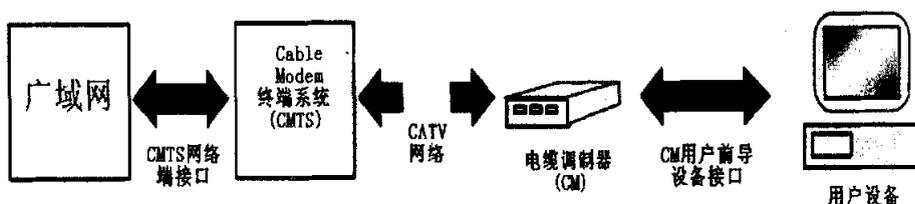


图3.1 DOCSIS网络系统架构图

CM系统的主要功能是在CMTS和用户之间透明传输数据包, 为此DOCSIS专门制定了RF接口的通信协议栈, 如图3.2<sup>[5]</sup>所示。物理层包括与实际物理媒质有关的子层 (PMD) 和下行传输汇聚子层 (Transmission Convergence)。物理层之上的数据链路层包括媒质接入控制层 (MAC)、链路安全子层 (Link Security) 和逻辑链路控制子层 (LLC)。MAC层定义的规程和参数使CM和CMTS能够相互作用, 以便于下行和上行信道能够支持各种业务。链路安全子层提供IP包传输所需的安全性。DOCSIS的规范主要是关于物理层和MAC层, MAC层以上的各层均采用现有网络体系结构的协议, 如TCP/IP协议簇。所以高层的协议和规范并不是DOCSIS规范关心的问题, 它需要解决的是如何在HFC网络特有的网络拓扑结构上的通信问题。

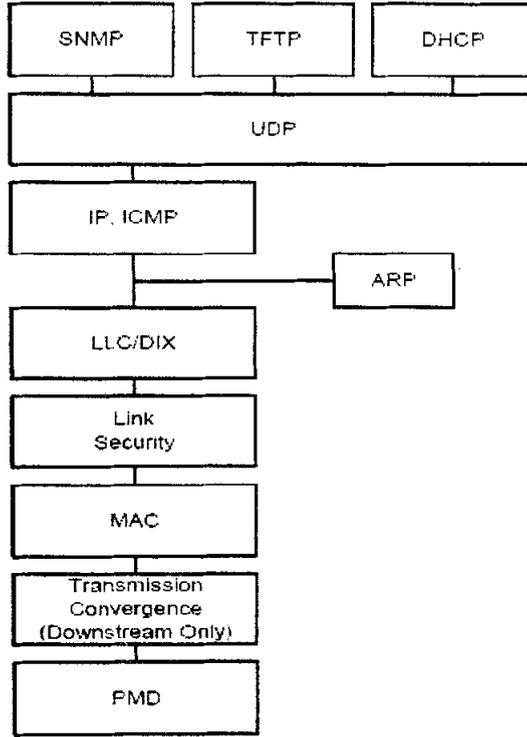


图3.2 RF接口的DOCSIS通信协议栈

图3.3<sup>[5]</sup>所示即为CM和CMTS的数据传送。需要特别注意的是，MAC层里上行和下行信道之间的数据传送和传统的局域网(LAN)不同，因为上行信道是在共享媒质上进行点到点的传输，而下行信道是在共享媒质上进行点到多电的传输。

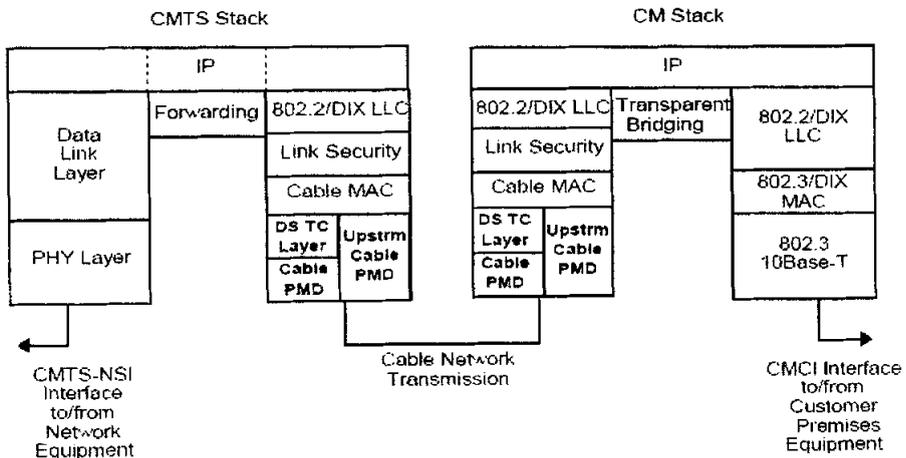


图3.3 通过CM和CMTS进行数据的传送

下面的章节将对DOCSIS协议比较重要的几个部分作介绍。

### 3.2. 物理层

CATV网络原来是为传输单向的模拟信号的电视节目而设计的, 原来的单向的模拟信号的传输频率是从50MHz到450MHz。由于采用HFC后, 传输的频率已经提高到750MHz或更高(1GHz), 这样就为双向业务的开展提供了便利的条件。DOCSIS协议中规定: 5MHz到42MHz用于上行信道的传输, 50MHz到450MHz用于模拟信号的电视节目传输, 450MHz到1000MHz用于下行数字信号的传输, 包括数据业务和数字电视节目等, 如图3.4所示。同时, 由于原有的模拟信号是按照每6MHz划分为一个频道, 所以HFC宽带接入网的下行方向也按照6MHz划分为一个信道。

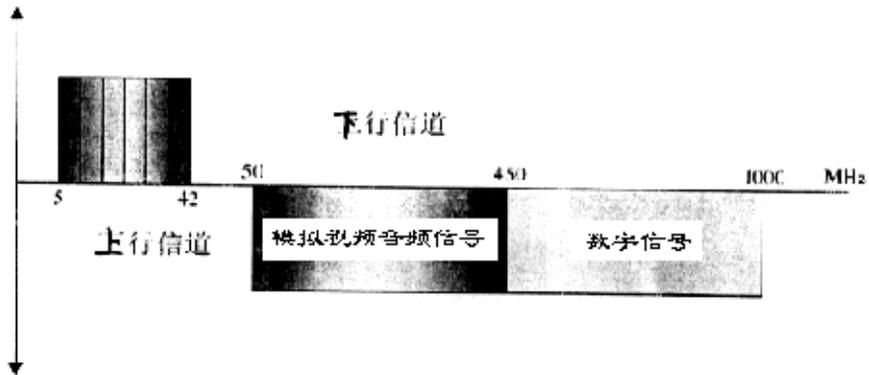


图3.4 HFC网络频率分配

物理层规定了前向纠错(FEC)采用R-S(Reed-Solomon)编码, 下行信道采用广播的通信方式, 调制方式为64QAM或者256QAM, 上行调制方式为16QAM或者64QAM。需要特别注意的是, 由于HFC网的特殊性, 即多用户共享上行信道, 所以上行信道按照FDMA(频分多址)被划分成多个子信道, 而每个子信道内采用TDMA(时分多址)的接入方式。这成为了HFC网络有别与其他接入网的最大特点之一。

下行比特流被定义为长度为188字节的MPEG数据包, 其中包括4字节的包头和184字节的载荷。MPEG数据包用来承载DOCSIS MAC帧。MAC帧可以从MPEG数据包中的任意位置开始, 而且几个MAC帧可以同时包含在一个MPEG数据包中。

### 3.3. MAC 层

DOCSIS中的MAC层定义了CM和CMTS相互配合的进程和参数, 以支持下行和上行信道的各种业务。由于HFC网络的拓扑特点, 所以开发一个适合的MAC层协议就显得尤为重要, 同时也相对困难。下面将详细讲解MAC层的规定和运作。

MAC子层协议的主要特征包括:

- 在CMTS控制下的上行带宽分配;
- 上行数据流被分为许多的微小时隙;
- 基于竞争和基于保留的上行数据流传输机会的动态管理;
- QoS: 如数据包分类, 动态服务的建立等;
- 在数据链路层提供安全功能扩展;
- 支持较大范围的数据速率。

#### 3.3.1. MAC 层的重要定义

##### (1) MAC域 (MAC-Sublayer Domain)

MAC域是多个上行信道和下行信道的集合, 这些信道使用单个MAC分配和管理协议。它包括一个CMTS和多个CM: CMTS必须为所有的上行和下行信道服务, 而一个CM可以选择多个上行及下行信道, 但是一次只能接入一个上行和一个下行信道。MAC域定义了该MAC层的范围和CMTS对CM的MAC地址的管理。

##### (2) 服务流 (Service Flows)

服务流是DOCSIS MAC层协议的基础部分。服务流是MAC层传送的业务, 它在CM和CMTS之间提供数据包的单向传送。服务流还可以根据QoS的业务量参数对业务量进行整形、提出策略和优先级确定。

在MAC域中, CMTS用服务号SID (Service ID) 来标识每一个CM, SID既是一个设备标识号, 同时也是对服务级别的一种管理。而一个服务流号SFID (Service Flow ID) 定义了一个CMTS与CM之间的单向映射。上行带宽被分配给SID, CMTS可以给一个CM分配一个或多个SFID, 这种映射可以在CM向CMTS注册时协商, 或通过“动态服务建立”来实现。SFID的长度为32bit, SID

的长度为14bit, 但有时以16bits的形式传输。任何时候, 一个CM都必须有一个活动的上行服务流和一个活动的下行服务流, 它们被称为上行/下行基本服务流, 而分配了上行基本服务流的SID被称为基本SID。所有的SFID在MAC域内都是独一无二的。

(3) 上行传输机会间隔时间, 微时隙

微时隙 (Mini-Slot) 是上行传输机会间隔时间 (Upstream Intervals) 的单位, 一个微时隙等于2的n次幂乘以6.25微秒 (这里的n由UCD Message来定义), 而上行传输机会间隔时间包含整数个微时隙。

(4) MAC帧 (MAC Frame)

MAC帧如图3.5<sup>[5]</sup>所示, 包含一个MAC帧头 (Header) 和变长数据PDU。MAC帧是CMTS和CM在MAC子层传输数据的基本单元 (包括上行及下行)。MAC的帧头唯一标识了MAC帧的内容, 它支持四种不同类型的数据PDU: 可变长度数据包、ATM数据包、MAC层专用数据包 (如请求、测距、带宽分配信息等)、预留的PDU。

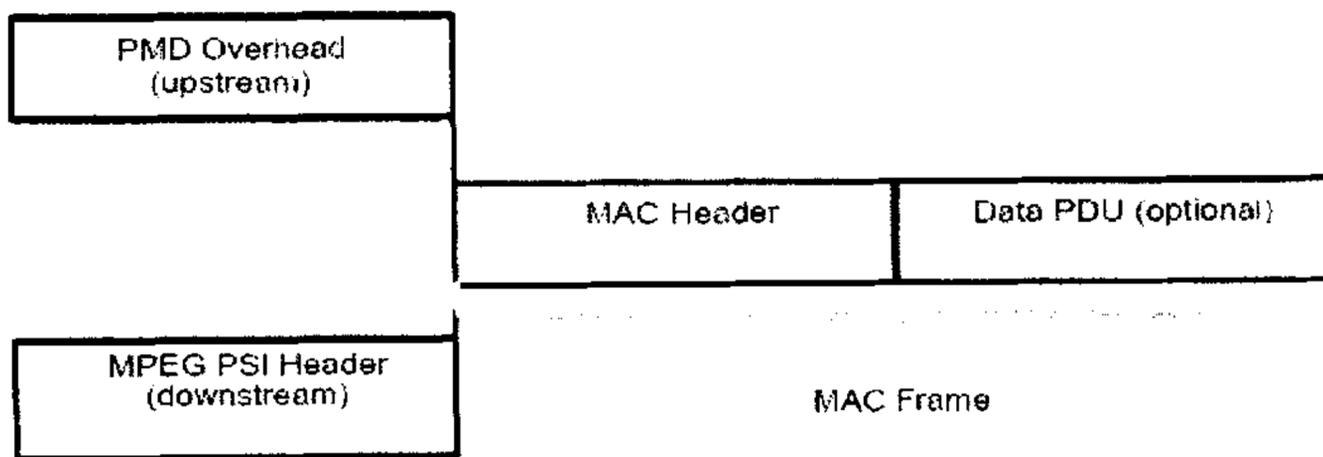


图3.5 MAC帧格式

### 3.3.2. 上行信道带宽分配

由于采用时分复用, 所以上行信道是以时隙 (slot) 的形式传输数据的。CMTS必须为这些时隙制定相对应的时间, 并且要能控制CM对这些时隙的使用。举个例子, CMTS可能会准许一些连续的时隙给一个CM, 以让此CM可以传输一个数据PDU。CM会依CMTS所指定的时隙来计算传输的时间, 而CMTS会在它所定义给CM的相对时间得到CM所传送的数据PDU。

在上行信道带宽分配机制中, 最基本的就是利用MAP, 如图3.6<sup>[5]</sup>所示。

MAP 是一个靠 CMTS 利用下行信道传输的 MAC 管理讯息 (Management message)。在 MAP 中, 明确指定了每一个时隙的使用者和用途。比如, 描述哪些时隙是用来让某些特殊使用者 (特定的一个或者几个 CM) 传输数据, 哪些时隙是用来让有上行传输需要的 CM 发出传输请求之用, 哪些时隙是用来让新加电的 CM 加入网络之用。

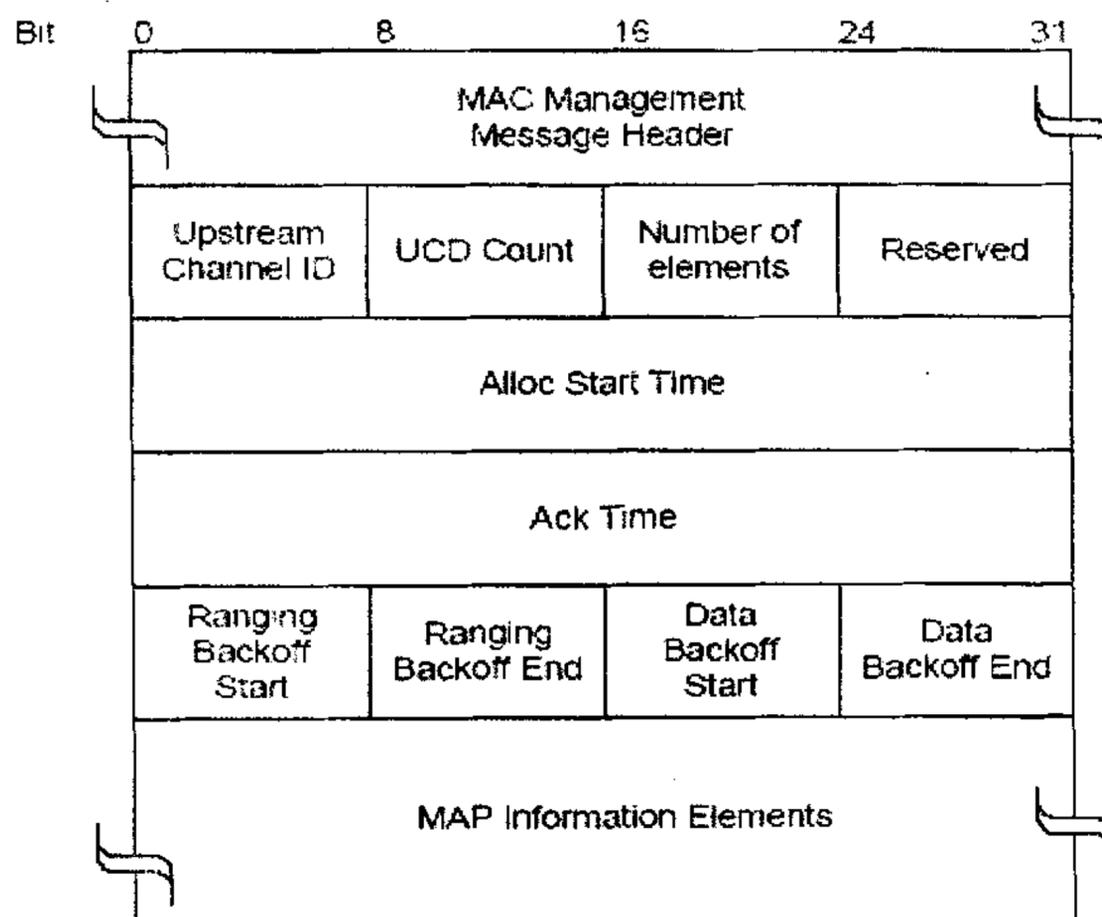


图3.6 MAP帧格式

MAP 帧中包含了固定长度的帧头和不同数量的信息元素, 叫作 IE (Information Elements), 每一个 IE 定义了一定范围时隙的使用。每一个 IE 是由一个 14bit 的 SID, 一个 4bit 的 Type Code 和一个 14bit 的 Starting Offset 所组成。其中用来为有需要的 CM 发出它的传输请求的 IE 叫做 Request IE。所有的 CM 必须扫描 MAP 中所有的 IE 来获知上行信道是如何分配的。而 CMTS 不是直接给出每个 IE 管理的时隙的时间范围, 而是通过 IE 的 Starting Offset 字段, 以 mini-slot 为单位, 给出一个相对于该 MAP 的有效开始时间的一个偏移量。IE 在 MAP 中的顺序要严格遵守各种 IE 的 Starting Offset 顺序。考虑到网络延时, CMTS 需要在 MAP 帧的有效时间 (即 MAP 帧中的 “Alloc Start Time”) 之前发送。每个 MAP 所描述的 mini-slot 个数都可以不相同。一个 MAP 最少可以描述一个 mini-slot, 最多可以描述 240 个 IE, 每一个单独的 IE 最大可以描述 256 个 mini-slot, 而它们所描述的

总的mini-slot不能超过4096个。

因此, 上行信道带宽分配机制所包含基本的要求总结如下:

(1) 每一个CM都有一个14bit的SID, 以及一个48bit的MAC地址。并且每一个CM都可以向CMTS发出带宽分配申请。

(2) 上行信道可以被切割成由多个时隙所组成的数据流。而时隙的编号是由CMTS来管理, 时隙的时间关系是由同步(SYNC)数据包来告知给所有的CM。

(3) 当CM需要带宽时, 必须要向CMTS提出带宽的要求。

(4) CMTS会在Allocation MAP帧中, 定义所有准许被使用的时隙, 并以下行信道来传输这些Allocation MAP帧。

下面以举例详细说明当CM有数据请求时与CMTS的交互运作, 如图3.7<sup>[5]</sup>。

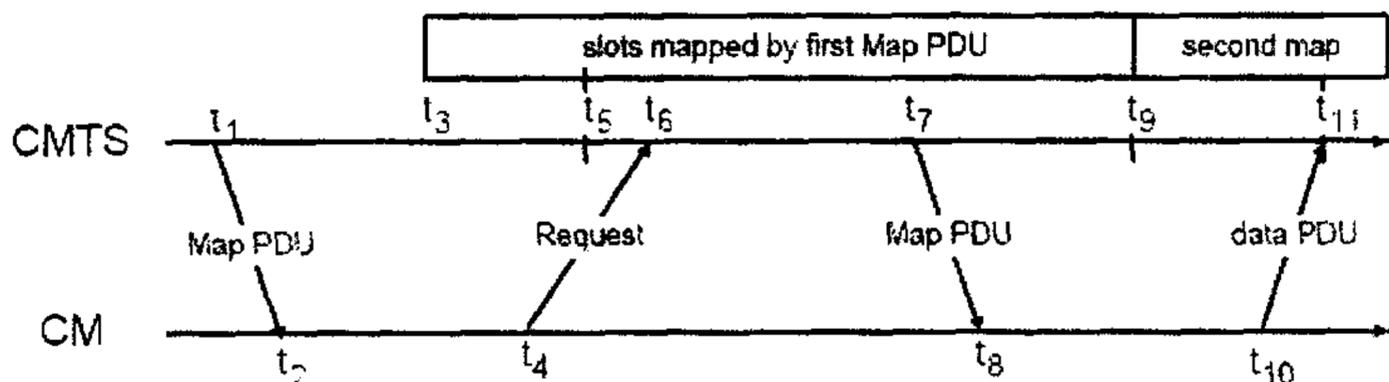


图3.7 CM与CMTS的运作

(1) 在 $t_1$ 时刻 (CMTS实际发送MAP帧的时刻), CMTS发送一个MAP给CM, 这个MAP的有效开始时间是 $t_3$ 。在这个MAP中存在着一个Request IE (即), 该IE分配给CM的发送时刻是 $t_5$ 。 $t_1$ 和 $t_3$ 之间的时间差, 主要基于以下几点考虑:

- 下行传播时延, 以便于CM能接收到此MAP;
- CM对MAP的处理时间: 比如CM对收到的MAP的解析, 并从中得出传输机会;
- 上行传播时延, 以便于CM发送的首个上行数据能够在 $t_3$ 时刻到达CMTS。

(2) 在 $t_2$ 时刻, CM接收到CMTS发出的MAP, 然后CM对其进行解析并寻找传输机会。为了降低冲突概率, CM选择 $t_6$ 时刻作为其发送。之所以选择 $t_6$ 时刻则是基于“截短的二进制指数回退算法”, 将在后面讲解。

(3) 在 $t_4$ 时刻, CM发送出请求帧。在该帧中, 包含了CM为了发送自己的数据PDU所需要的mini-slot个数。为了补偿HFC网络的延时, 使得对于CMTS来

讲, 它所规定的CM的发送时刻即等于CMTS的接收时刻, 所以CM必须提前发送数据包, 这个提前量是由初始的定时和测距决定的。这里, CM就必须在提前于 $t_6$ 时刻的 $t_4$ 时刻发送, 以使该请求帧能够在 $t_6$ 时刻准时达到CMTS。

(4) 在 $t_6$ 时刻, CMTS接收到CM发出的请求帧, 然后在下一个MAP中为其分配带宽。但是, 在实际情况下, CMTS将不一定能够满足这个CM的上行信道带宽申请, CMTS究竟为哪一个CM分配带宽, 这取决于CMTS采用的分配算法。

(5) 在 $t_7$ 时刻, CMTS给它将要为其分配带宽的CM发送下一个MAP, 该MAP的有效开始时间是 $t_9$ , 在这个MAP中, 将包含一个Short/Long Data Grant IE, 这个IE规定CM在 $t_{11}$ 时刻开始发送数据PDU。

(6) 在 $t_8$ 时刻, CM接收到CMTS发给它的带宽分配MAP。于是, 它对其进行解析并寻找授权给它的发送机会。

(7) 在 $t_{10}$ 时刻, CM发送它的数据PDU。之所以选择 $t_{10}$ 发送, 也是跟步骤3中的原因一样, 为了补偿上行延时, 以便于这个数据PDU能够在CMTS规定的 $t_{11}$ 时刻到达CMTS。

需要特别注意的是:

(1) 由于HFC网络的特殊性, 在步骤3中, 可能由于冲突而使CMTS不能正确接收CM发送的请求帧。但是CMTS并不直接检测是否发生了冲突, 而只关心正确收到的请求。对于CM而言, 如果接收到的下一个MAP中不包含对它的带宽请求的确认, 于是冲突就发生了。此时, CM必须执行截短的二进制指数回退算法并进行重试。

(2) 在步骤4, 有可能即使CMTS收到了CM发送的上行带宽请求帧, 但由于其它原因, 它还是不能给该CM分配一定的上行时隙。这时, CMTS要么给相应的CM回复一个MAP, 只不过在这个MAP中, 它分配给CM一个上行时隙数为零的授权; 要么就干脆扔掉CM的请求帧。如果CMTS回复给CM一个具有零长度的授权, 那么它在后续的MAP中都必须继续发送上行时隙数为零的授权, 以通知CM, 它的请求正在被挂起等待处理, 直到CMTS能够给相应的CM分配一定的带宽, 或者CMTS决定扔掉CM的这个请求帧。同时, 对于CM来讲, 只要它在接收隙数为零的授权, 那么它就不再为前一个服务所需要上行带宽发送新的请求帧。

由上面的分析, 可以将所有的时隙分成两大部分: 请求时隙和数据时隙。由于请求时隙可能存在冲突而必须采用竞争的方式 (将在下面讲解), 所以也叫作竞争时隙。因此, 上行信道的带宽分配可以大体被划分成两个部分:

- (1) 竞争时隙和数据时隙的分配;
- (2) 数据时隙内的分配, 即服务于哪个用户, 如何服务。

### 3.3.3. 上行传输的冲突解决

根据前面分析的分析, 当CM有数据传输的时候, CM就会根据最近的一次接收到的MAP中定义, 在规定的竞争时隙内发出请求, 但是由于上行信道是多用户共享信道, 所以可能会发生多个用户同时在一个竞争时隙内发出请求, 这样就产生了冲突。

为了解决冲突, DOCSIS使用的方法是“截短的二进制指数回退算法”。在这里我们将回退的值抽象为了一个窗口, 这个回退窗口的初始值与最大值都由CMTS在MAP中的“Ranging Backoff Start”与“Ranging Backoff End”字段进行控制, 这两个字段都表示2的指数, 如该字段的值为4, 则表示窗口为0~15, 同样, 字段的值为10则表示窗口为0~1023。

每当CM想要在竞争环境中发送数据时, 它都要在它的回退窗口中随机地选择一个数值。这个数值表示了CM在发送数据之前必须等待的传输机会的个数。一个传输机会是指一个请求所需的时隙个数, 为了讨论的方便, 以下都将一个传输机会认为是一个时隙, 所以这个随机选择的数值也就是CM在当前规定窗口内必须回退 (即等待) 的竞争时隙个数。例如, 假设一个CM的初始化回退窗口是0~15, 而它随机地选择了一个数值11, 则该CM都必须把它的发送推迟11个传输机会 (时隙)。如果CMTS分配给该CM的第一个可用的Request IE给它提供了6个竞争时隙, 那么该CM将不使用这6个时隙, 并且还剩5个竞争时隙需要等待, 如果该CM的第二个Request IE给它提供了9个竞争时隙, 则它将再延迟6个, 并在第7个竞争时隙发送请求。

在请求发出之后, CM会等待下一个MAP中所描述的数据准许(Data Grant)或数据确认 (Data Acknowledgement)回复, 一旦得到上述任何一个回复, 则CM的本次请求竞争成功。但是, 如果CM接收到的下一个MAP中, 不存在Data Grant或者Data Acknowledge中的一个, 并且这个MAP的Ack time字段的值比CM的最

近一次的发送时间还晚时, CM便认为竞争发送失败, 即发生了冲突。

当CM发现传输出现了冲突后, 它将以2的倍数来增加它的初始回退窗口(但是扩大后的窗口必须小于CMTS规定的最大回退窗口), 然后CM再随机地在它的新的初始回退窗口中选择一个数值, 重新执行上述的操作, 如果重试次数超过16次, 则CM将删除该数据请求。

如果CM在延迟发送期间接收到了给它的明确的(以与它前面所发送的请求帧的SID相对应的) Data Grant, 那么CM将中止竞争状态并利用这个显式的发送机会发送数据。

CMTS 对频宽需求机制的控制是很有弹性的, 在某种情形, CMTS可以仿照以太网来设定Ranging back-off start和Ranging back-off end, 但也可以考虑到公平性和效能的因素, 在MAP中, 将Ranging back-off start和Ranging back-off End设置为一个固定值。同时, CMTS也可以经常更新这两个值的定义, 以使CM得到一个最佳化的使用效能。

#### 3.3.4. CMTS 与 CM 之间的时隙同步

设计 DOCSIS 标准中 MAC 层协议的一个主要挑战是如何解决 HFC 网络中的巨大延时。研究<sup>[8]</sup>表明, HFC 网络中的延时可达 400us~800us。CM 为了能精确地在 CMTS 为其分配的时隙内传输数据, 必须要设法补偿这些延时。DOCSIS 标准的 MAC 层规定 CM 通过以下两个信息来实现与 CMTS 的时隙同步。

(1) 全局定时参考。这个值由 CMTS 产生, 并以 SYNC 帧格式通过下行信道发送给每个 CM。SYNC 帧中包含了一个 32 位的 mini-slot 计数值, 该值表示该 SYNC 帧的发送开始时隙。当 CM 接收到 SYNC 帧后, 比较实际收到时间与该 SYNC 帧的实际发送时间, 并根据比较结果相应地调整自己的本地定时参考。

(2) 定时偏移量。偏移量的值由 CM 通过初始化测距与周期性测距过程得到。测距的目的之一是计算 CM 与 CMTS 之间的往返延时, CM 通过提前发送时间以补偿这个延时, 以达到与 CMTS 的时隙对齐。测距的另一个目的是 CMTS 调整 CM 的发射功率等工作参数。

CM 上电之后, 首先从下行信道获得 SYNC 帧, 以校准本地定时参考。由于网络传输时延的存在, CM 校准后的本地定时参考将会有有一个时延  $t$ 。然后

CM 在收到的 MAP 帧中寻找初始测距时隙,并在其中发送请求帧(RNG-REQ)。CMTS 在接收到 CM 的 RNG-REQ 帧后,根据其到达时间与在 MAP 帧分配的 RNG-REQ 帧发送时间,计算出相应的时间调整量,同时测量 CM 的发射功率、载波频率等,并等到相应调整值,然后以测距响应帧(RNG-RSP)格式发送 CM。

CM 根据 RNG-RSP 中的信息调整自己的发射功率、频率等工作参数,并把其中的时间调整量作为提前发送量,在 CM 下次发送时,它将通过提前发送以补偿网络延时,实现与 CMTS 的时隙同步。

CM 通过初始化测距之后,可以实现与 CMTS 的时隙同步。为了跟踪网络延时的变化,CM 除了初始化测距外,还需要周期性进行测距,以维持与 CMTS 的时隙同步。周期性测距与初始化测距的不同之处在于,前都是在属于某个 CM 的单播时隙内进行,而后者是在竞争时隙中完成。

### 3.4. 小结

本章分析了 DOCSIS 标准的基本架构,以及物理层和 MAC 层的定义。重点研究了上行信道的带宽分配和冲突解决是如何运作的。需要注意的是,DOCSIS 标准对许多问题的解决并没有做出具体的规定。其中,上行信道的带宽分配和冲突解决这两个关键的算法方案留给了研究者去自己设计。这就成为了现在世界上 HFC 宽带接入网的研究热点,也是本文研究的重点所在。

## 第4章 基于冲突检测的竞争解决算法研究

### 4.1. 研究问题分析

以HFC网络作为宽带接入网,有其自身的优势,比如带宽资源丰富等。但是,HFC宽带接入网作为一种较为特殊的网络结构有一些与传统网络明显的区别,同时,这些区别也带来一些问题,这也正是HFC宽带接入网的设计难点所在。其中的两三个主要问题是:双向改造、噪声和检测冲突。

(1) HFC网络最初是为传输单向的广播信号设计的,网络不具备双向通信的功能。为了能在HFC网络上开展双向通信业务,必须对它进行双向改造。但这些年,双向改造的进展一直比较缓慢,究其原因,主要是投资巨大。据初步估计,双向改造的投资至少是铺设单向网络投资的两倍多。<sup>[10]</sup>美国AT&T公司进行的双向改造,平均在每个用户身上花费多达2300美元,如此大规模的投资是否能回收、如何回收是有线网络商需要重点考虑的问题。从20世纪90年代末期开始,中国相继有一些城市进行双向HFC网络改造,但一直进展不大,目前总体宽带用户也不过30万。目前,在上海、深圳等少数经济发达地区,HFC网络双向改造较早,覆盖面较大,加以时日,回收成本还不是太大的问题;而在更多的城市,网络双向改造较晚或还没有改造,特别是中西部地区。而且,电信的ADSL和以太网为主的小区宽带早已开始推广市场,HFC宽带接入网必然要与它们竞争以争夺客户,HFC网络作为一种新的宽带接入网在市场占有率上明显处于劣势。

因此,HFC网络的双向改造面临很大困难,现阶段中国大部分的HFC网络都还没有进行全面的双向改造,甚至在许多地区,HFC网络双向改造的计划还遥遥无期。

(2) HFC网络用于传送上行信号的频段一般为5~40MHz,由于HFC网络的同轴电缆部分是树形结构,所以上行信道的噪声是各支路噪声和各支路放大器的级联噪声叠加,即“漏斗效应”,使回传噪声累积到前端,严重恶化了上行信道的信噪比。

这些噪声通常包括入口噪声、共用通道的失真、激光器发射机和光接收机

的噪声<sup>[3]</sup>。其中入口噪声是最主要的, 它分为窄带干扰、脉冲干扰和突发噪声。窄带干扰是指上行频带内先行指配给各种无线电业务的频率所形成的干扰。这些干扰源的频率及场强因地而异, 所以只能在上行信道安排中避开经测定的窄带干扰频率, 但会影响上行信道利用率。脉冲干扰指用户端周围的工业电器和家用电器所形成的宽带干扰。这些干扰也因地而异, 尤其是接入到系统的电视机的老化程度对回传噪声的影响, 将更难控制。避免干扰的途径应采取加强系统屏蔽隔离的方法。例如, 所采用的同轴电缆、接头等要求有较高的屏蔽性, 终端接入盒要插入滤波器的隔离, 但是这要付出经济代价。突发噪声是认为或自然产生的噪声源。

随着用户数目的增加, 入口噪声电平也急剧增加。噪声的“漏斗效应”使 HFC 网络的上行信道的抗干扰、抗噪声问题成为了一个世界性的难题, 长期得不到有效地解决。

(3) 正如第三章的论述, HFC 宽带接入网的上行信道竞争冲突的研究是一个关键点也是一个难点。究其原因, 这是由于 HFC 网的自身特点所决定的。在 HFC 宽带接入网中, 由于树形的拓扑结构, 每一个用户都不能像其他共享网络中那样首先去检测共享信道是否正在被使用, 而且, 用户之间彼此独立, 它们也不能互相通知协调<sup>[13]</sup>。在类型上, 这属于“非载波侦听 (no carrier sense)”。DOCSIS 协议标准给出的解决办法是一种“请求/授权”机制。也就是说, 对于 CM 而言, 当它有数据需要传送的时候, 不能直接发送, 必须首先发送请求, 等待 CMTS 对它的这一请求的授权。而由于前面所述的原因, CM 只能比较“盲目”地发送自己的请求, 只有在其后等待 CMTS 授权的步骤中才能获知自己这一次的请求发送是否已经发生了冲突。为了解决这个问题, DOCSIS 引入了“截短的二进制指数回退算法”, 让 CM 随机地等待一定时间再发送请求。这一算法的基本思路是使多个 CM 对竞争时隙的使用分布更均匀, 较少地出现使用过于集中的情况。但是, 这一算法, 也只是规定了 CM (即用户端) 如何运作以尽量避免冲突, 并没有规定 CMTS 应该如何运作来控制窗口的大小以解决冲突。这也正是 HFC 网络设计的难点, 也是研究者研究的重点所在。

在共享网络中, 冲突是不可避免的, 所以人们早就开始研究如何解决多路访问信道的竞争和冲突问题, 并且已经有了许多成熟的思想、协议, 其中许多早已经推广使用。那我们能否从这些多路访问协议对竞争和冲突的解决思路中

得到一定的启发和借鉴呢? 下面简要分析几个常用的多路访问协议和DOCSIS协议之间的联系和区别。

## 4.2. DOCSIS 与常用多路访问协议联系区别的研究

### 4.2.1. 常用多路访问协议分析

目前, 已有许多广为人知的多路访问信道的分配算法, 这里简要分析其中几种与我们研究相关的协议。

#### (1) ALOHA

ALOHA协议的基本思想是用户有数据要发送时, 可以直接发至信道; 然后监听信道看是否产生冲突, 若产生冲突, 则等待一段随机的时间重发。<sup>[1]</sup>它的目的是解决信道的动态分配, 基本思想可用于任何无协调关系的用户争用单一共享信道使用权的系统。它的一种改进是分时隙ALOHA: 将信道时间分成许多时隙, 每个时隙对应一个帧。系统中的所有用户都必须在时隙的开始时刻发送数据帧。主时钟的同步信息要向所有用户广播, 以使网内所有用户都与主时钟同步。

#### (2) CSMA

载波监听多路访问CSMA (carrier sense multiple access)<sup>[1]</sup>, 也称作“先听后说”, 是由ALOHA改进而成。它的基本思路和ALOHA一致, 关键的改进在于冲突发生后的运作。要传输数据的站点首先对信道上有无载波进行监听, 以确定是否有别的站点在传输数据。如果信道空闲, 该站点便可传输数据; 否则, 该站点将避让一段时间后再做尝试。这就需要有一种退避算法来决定避让的时间, 常用的退避算法有非坚持、1-坚持、P-坚持三种。

#### (3) CSMA/CD

载波侦听多路访问/冲突检测协议 (CSMA/CD) 又是对CSMA的改进。<sup>[1]</sup>在CSMA中, 由于信道传播时延的存在, 即使总线上两个站点没有侦听到载波信号而发送帧时, 仍可能会发生冲突。由于CSMA算法没有冲突检测功能, 即使冲突已发生, 仍然将已破坏的帧发送完, 使总线的利用率降低。

CSMA/CD的改进方案是使发送站点传输过程中仍继续侦听信道, 以检测是否存在冲突。一旦检测到冲突, 就立即停止发送, 并向总线上发一串阻塞信

号,用以通知总线上其它各有关站点。这样,通道容量就不致因白白传送已受损的帧而浪费,可以提高总线的利用率。这种算法可以形象的概括为“先听后讲,边听边讲”,已广泛应用于局域网中。它的一个版本IEEE 802.3标准,即是以太网标准。

当冲突产生时,一般采用称作“二进制指数后退算法”来确定后退时间周期。方法是: $i$ 次冲突后,等待时隙数就从 $0 \sim 2^i - 1$ 中随机选出。最小后退周期至少是一个时隙。如果冲突次数高,表明网络繁忙,发送站应增加后退时间,以降低再次产生冲突的可能性。当达到10次冲突后,随机等待的最大时隙数就被固定在1023。16次冲突后,控制器将不再动作,并向计算机报告发送失败。

#### 4.2.2. 联系和区别研究

从上面的分析可以看出,DOCSIS协议和这些常用的多路访问协议有许多相通之处。而且,在DOCSIS的实际制定过程中,确实从它们身上充分借鉴了许多有价值的思路。

首先,DOCSIS采用了和ALOHA相同的分时隙的信道分割方法。所有的站点都在同一个时钟下,按照时隙的规格发送数据帧或者延迟等待。其次,DOCSIS采用了和CSMA/CD几乎一样的回退算法。只是因为第一次回退不是从0和1个时隙开始,所以才被称为“截断的二进制指数后退算法”。

但是,DOCSIS协议和这些常用的多路访问协议也有非常大的区别,主要集中在对冲突的检测上。在这些常用协议中,站点在冲突检测过程中扮演了一个非常重要的角色。其根本原因在于这些系统中没有中心控制器,是一种完全分散的控制系统。所以,站点会自身使用一种能及时判断冲突的方法去主动地检测,常用的是对信道载波的侦听。而在DOCSIS协议中,站点对于冲突的检测就要慢了许多。根据上一章的分析,每个用户站点在发送前都不会去检测信道是否被利用,而是经过等待收到CMTS的授权信息后,再在其中寻找是否有回复的信息。如果没有,则它判断上一次的发送发生了冲突。这种检测冲突的办法非常盲目,比起上面那些协议能够“先听后讲,边听边讲”,效果自然就差了许多。

那我们是否能仿效其他协议,在DOCSIS协议中也引入用户站点主动检测信道从而判断冲突这一个思路呢?很遗憾,这在HFC网络上是无法实现的。因

为HFC网络是一个树形的拓扑结构，而且上行和下行通道是分开的，所以对于上行而言，每一个CM在自身所在的那一个支路上无法获知其他CM所在的支路和共享信道的情况。但是，这并不意味着我们就只能放弃这一思路。如果我們能在HFC网络中，以其他方式来实现对信道的主动检测，从而实现对冲突的主动检测，这将极大地改善原有协议中对冲突的解决，进而有效提高整个网络运作的效率。下面的章节就将详细论述这个问题。

### 4.3. 对 HFC 宽带接入网的改进

#### 4.3.1. 无线传输的上行信道

在我们的本次研究项目中，提出了利用无线信道来替代HFC网络的上行信道，如图4.1所示。下行信道依然采用HFC网络，而上行信道使用无线传输的方式。由于只是物理层的改变，所以DOCSIS协议中关于MAC层以上的协议依然适用。而现在无线传输的技术已经非常成熟了，有许多成熟的模块可以直接使用；同时，也有许多公共的频段可以利用，如2.4GHz<sup>[11]</sup>。更重要的是，使用无线传输的方式可以解决前面分析的三个问题。

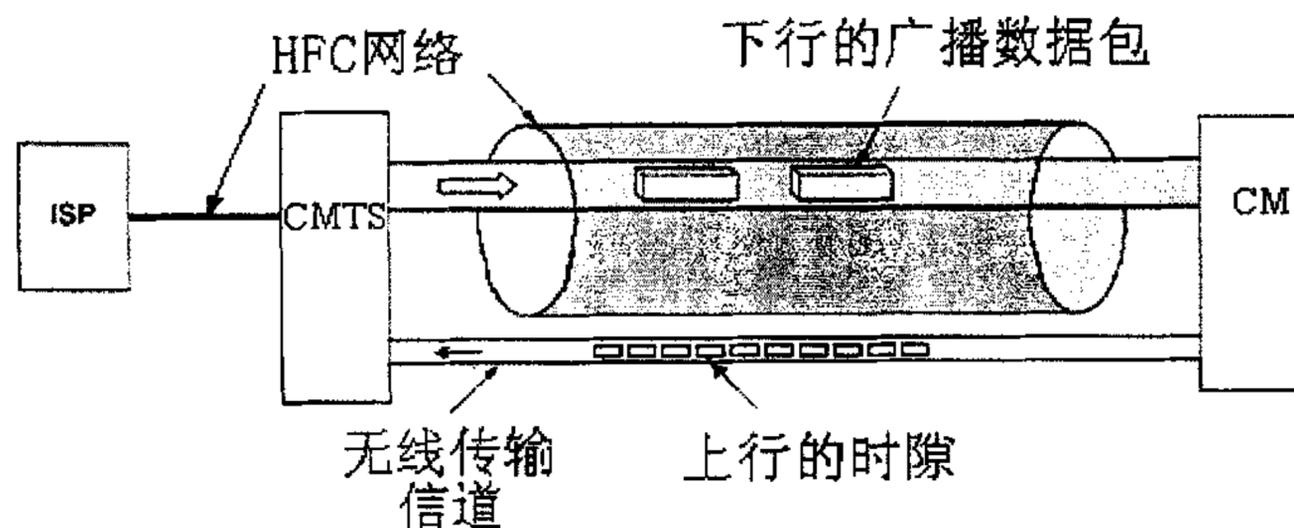


图4.1 无线上行信道的HFC宽带接入网

(1) 由于只是在CM和CMTS上加装无线的传输模块，所以根本不需要对现有的HFC网络进行太多的双向改造。因此，摆在推广HFC宽带接入网面前最大的困难：双向改造的难度和昂贵，也就迎刃而解了。只要是在有HFC网络的地方就可以直接安装CMTS和CM设备，从而马上开展宽带接入服务。而中国目前的HFC网络普及率非常高，这样就极大地有利于HFC宽带接入网的发展。

(2) 使用无线传输后, 就避免了许多原来在HFC网络上会被引入的噪声。同时, 也就解决了HFC网络中噪声级联叠加而产生的“漏斗效应”。

(3) 既然由前面分析的, 用户端 (CM) 无法实现对信道的检测, 那我们是否可以换一种思路, 能否在CMTS上实现信道检测呢?

CMTS作为一种DOCSIS协议的HFC宽带接入网设备, 有其独特性。在其他共享网络结构中, 用户站点有很大的控制能力。比如以太网中, 站点可以自己主动侦听网络决定是否发送, 何时发送。而在HFC宽带接入网中, CM (即用户) 的控制权限非常小, 大部分的控制都集中在了一个非用户端 (头端) 的控制设备上, 即CMTS。所以, CMTS在HFC网络中处于中心地位, 它要控制大部分的网络运作, 而不只是像以太网中的集线器或者交互机那样完成转发的功能。<sup>[12]</sup> 因此, 如果能将信道检测的功能从用户端转移到CMTS上, 那么就可以利用这一功能来协助完成对网络运作的控制。下面一节就论述这个问题。

#### 4.3.2. CMTS 对上行信道的检测

这里, 我们直接利用无线传输模块的特点来实现CMTS对上行信道的检测, 其基本思路还是对于信号载波的检测。

在无线传输模块中, 由于无线信号传输具有高噪声的特点, 所以必须有自动增益控制 (AGC, Automatic Gain Control) 模块。当检测到有信号载波以后, 就根据信号强度自动调整增益来放大信号, 这时就会有自动增益电平输出。我们就利用这个电平值, 一旦无线模块检测到信号载波, 就意味着有信号输入, 即有数据包到达头端。CMTS就会将数据包解调、解码, 一旦解码失败, 就说明这是一个被冲突的无效的数据包。根据之前时钟同步机制, 如果这个数据包是在某个竞争时隙的时间范围内接收到的, 那就说明该竞争时隙发生了冲突。同理, CMTS还可以检测出哪些竞争时隙被成功使用, 哪些未被使用。

现有的无线传输模块都可以比较轻易地检测自动增益电平。比如, 有的射频芯片直接提供引脚, 可以输出这个电平; 而有的将其放入内部的寄存器里, 可以供用户读取。这就为我们的改进设计提供了极大的方便。

#### 4.4. 基于冲突检测的竞争解决算法原理

根据前面的分析, 使用无线传输的上行信道, CMTS可以检测上行信道,

以获知时隙是否被使用。关于时隙，根据在3.3.2章节中的分析，在DOCSIS协议中，时隙被分成两部分：竞争时隙和数据时隙。数据时隙是CM按照自己的需要而申请用来传输上行信道数据的，所以不管CMTS分配了多少数据时隙，一定都会被使用。

但是，对于竞争时隙而言，情况就要复杂一些。由于CM执行“截短的二进制指数回退算法”，每个CM都随机地等待延迟一定的竞争时隙，也可以看成是CM从当前的规定窗口中随机地选中一个竞争时隙，如图4.2所示。

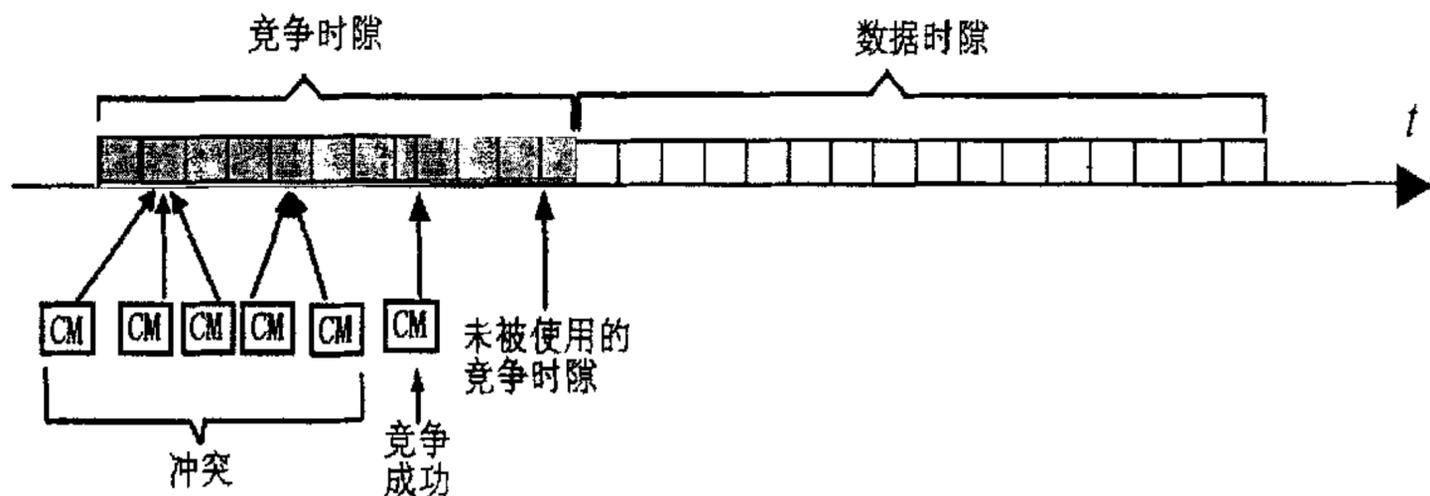


图4.2 CM对竞争时隙的使用

当一个竞争时隙只被一个CM选中的时候，这个CM就竞争成功，可以利用这个被自己独享的竞争时隙来发送请求信息。当有多个CM同时选中了一个竞争时隙时，冲突就产生了。同时，也可能有竞争时隙没有被任何CM选中，所以就产生了未被使用的时隙。所以，竞争时隙分成了“被使用的时隙”和“未被使用的时隙”，其中“被使用的时隙”又分成了“产生冲突的时隙”和“竞争成功的时隙”。

在CSMA/CD协议中，站点只要能发现冲突就可以采取措施，而不用去管冲突的程度到底如何。由上面的分析，CMTS能够从对上行信道的检测中发现有多少时隙产生了冲突。但是在DOCSIS协议的网络中，只是基于这点是不够的。因为冲突可能不光是由两个CM产生的，很可能是由多个CM产生的。CMTS不同于CSMA/CD协议中的站点，它是整个网络的控制中心，它必须对冲突的程度有一个比较准确的把握才能采取措施。

综上所述，基于冲突检测的竞争解决算法可以按流程顺序大致划分成三个部分，简要如图4.3所示。

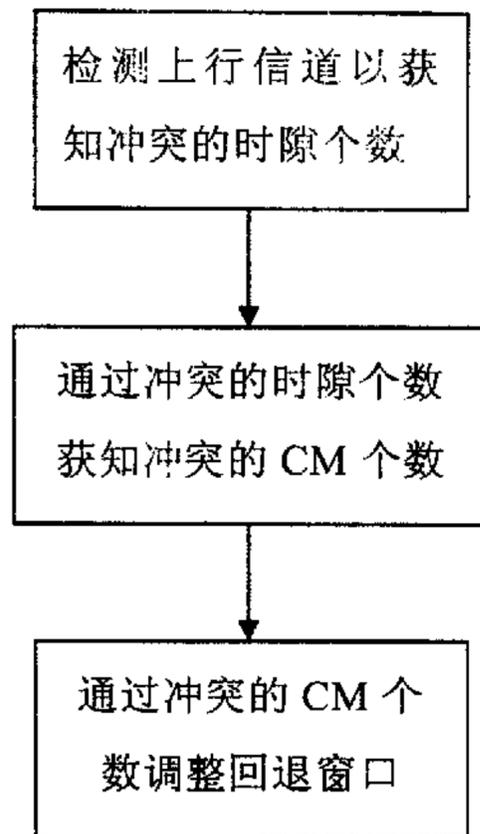


图4.3 基于冲突检测的竞争解决算法略图

下面的章节将详细分析和论述该算法的后面两个部分。注意，为了分析的需要，本章假设回退窗口的大小即为一个MAP帧中分配的竞争时隙个数，特此说明。

#### 4.4.1. 冲突 CM 的检测

CMTS只能直接检测到冲突的时隙个数，但是到底有多少CM导致了这些时隙冲突呢？一个直观的理解是，两个CM冲突一个时隙的概率应该比三个CM冲突一个时隙的概率要大，那能否认为一旦检测到一个冲突时隙就意味着有两个CM正在同时竞争这个时隙呢？

假设现在窗口大小为 $m$ ，即 $m$ 个竞争时隙， $n$ 个CM需要发送请求信息，所以数学模型就转化为： $n$ 个用户，每个用户都随机选中 $m$ 个时隙中的某一个。对于一个时隙而言，设 $x$ 为同时有 $x$ 个用户选中了这个时隙，则概率 $p(x)$ 表示一个时隙同时被 $x$ 个用户选中的概率，则：

$$p(x) = C_n^x \left(\frac{1}{m}\right)^x \left(1 - \frac{1}{m}\right)^{n-x} \quad (4-1)$$

其中 $\left(\frac{1}{m}\right)^x \left(1 - \frac{1}{m}\right)^{n-x}$ 表示有指定的 $x$ 个用户选中这一个时隙的概率， $C_n^x$ 表

示 $n$ 个用户中抽取 $x$ 个用户的组合个数。

当 $x=0$ 时,  $p(x)$ 表示没有用户选中这个时隙, 即该时隙没有被使用的概率。当 $x=1$ 时,  $p(x)$ 表示只有一个用户选中这个时隙, 即该时隙被成功竞争利用的概率。当 $x=2, 3, \dots, n$ ,  $p(x)$ 表示多个用户选中了这个时隙, 即该时隙发生冲突的概率。

$p(x)$ 这一概率会呈现如何的变化规律呢? 下面我们先用两个实际的例子计算来观察。

(1) 当 $n=4$ ,  $m=3$ 时:

<b>X</b>	0	1	2	3	4
<b><math>p(x)</math></b>	19.753%	39.506%	29.630%	9.877%	1.235%

(2) 当 $n=5$ ,  $m=3$ 时:

<b>X</b>	0	1	2	3	4	5
<b><math>p(x)</math></b>	13.169%	32.922%	32.922%	16.461%	4.115%	0.412%

现在将上面的数据重新整理, 假设发生了冲突, 下面所示即为每一种的冲突情况所占的百分比。

(1) 当 $n=4$ ,  $m=3$ 时:

冲突的CM个数	2	3	4
所占百分比	72.7%	24.3%	3%

(2) 当 $n=5$ ,  $m=3$ 时:

冲突的CM个数	2	3	4	5
所占百分比	61.1%	30.5%	7.6%	0.8%

观察上面的两组数据, 我们发现  $p(x)$  随着  $x$  的变化, 都呈现出先递增后递减的变化趋势。当  $n=4$ ,  $m=3$  时, 当  $x=1$ , 即只有一个用户选中的概率是最大的, 其后概率开始逐渐变小; 而当  $n=5$ ,  $m=3$  时, 有一个和两个用户选中的概率是最大的, 然后变小。对于  $x$  等于或大于 2 的情况, 即发生冲突的概率, 在这两组数据中, 都是两个用户冲突的概率最大, 但是三个用户冲突的概率为两个的 30%~50%。

如果式 (4-1) 的变化都是如上面数据显示的那样, 只有一个最大值, 那么我们就可以找到在一定的  $n$  和  $m$  下,  $x$  取值为多少的时候概率最大。即求出在一定情况下, 一个时隙被多少个用户同时选中的概率是最大的。下面对式 (4-1) 求导, 判断是否存在极值和几个极值。

令  $f(x) = p(x)$ , 将式 (4-1) 展开并且取自然对数:

$$\because C_n^x = \frac{n!}{x!(n-x)!}$$

$$\begin{aligned} \therefore f(x) = p(x) &= C_n^x \left(\frac{1}{m}\right)^x \left(1 - \frac{1}{m}\right)^{n-x} \\ &= \frac{n!}{x!(n-x)!} \left(\frac{1}{m}\right)^x \left(1 - \frac{1}{m}\right)^{n-x} = \frac{n!(m-1)^{n-x}}{x!(n-x)!m^n} \end{aligned}$$

$$\begin{aligned} \therefore \ln f(x) &= \ln \left( \frac{n!(m-1)^{n-x}}{x!(n-x)!m^n} \right) \\ &= \sum_{i=1}^n \ln i + (n-x) \ln(m-1) - \left[ \sum_{k=1}^x \ln k + \sum_{p=1}^{n-x} \ln p + n \ln m \right] \end{aligned}$$

其中  $\sum_{i=1}^n \ln i$ ,  $\sum_{k=1}^x \ln k$ ,  $\sum_{p=1}^{n-x} \ln p$  都表示从 1 开始的自然对数的连加, 虽然所用

符号不一致, 但不影响化减, 所以上式可化为:

$$\ln f(x) = \sum_{i=x+1}^n \ln i - \sum_{p=1}^{n-x} \ln p + (n-x) \ln(m-1) - n \ln m \quad (4-2)$$

令  $g(x) = \sum_{i=x+1}^n \ln i - \sum_{p=1}^{n-x} \ln p$ , 则式 (4-2) 变为:

$$\ln f(x) = g(x) + (n-x) \ln(m-1) - n \ln m \quad (4-3)$$

由于  $n$  和  $m$  为常数, 所以对 (4-3) 求导得到:

$$[\ln f(x)]' = \frac{1}{f(x)} f'(x) = g'(x) - \ln(m-1)$$

$$\therefore f'(x) = [g'(x) - \ln(m-1)]f(x) \quad (4-4)$$

$$\text{又} \because f(x) = \frac{n!(m-1)^{n-x}}{x!(n-x)!m^n}, \quad n, m, x \text{ 皆为整数}$$

$$\therefore f(x) \neq 0$$

为了找  $f(x)$  的最值, 所以令  $f'(x) = 0$ , 得到:

$$f'(x) = [g'(x) - \ln(m-1)]f(x) = 0$$

$$\therefore g'(x) - \ln(m-1) = 0$$

$$\text{即 } g'(x) = \ln(m-1) \quad (4-5)$$

于是式 (4-1) 是否存在极值的问题就转化为, 是否存在  $x$  使得式 (4-5) 成立。对式 (4-5) 等号两边同时积分以消掉求导, 并将  $g(x)$  展开为原形, 得到:

$$\int g'(x)dx = \int \ln(m-1)dx$$

$$g(x) = \ln(m-1)x$$

$$\therefore \sum_{i=x+1}^n \ln i - \sum_{p=1}^{n-x} \ln p = \ln(m-1)x \quad (4-6)$$

$$\text{令 } y = \sum_{i=x+1}^n \ln i - \sum_{p=1}^{n-x} \ln p - \ln(m-1)x, \text{ 由式 (4-6), 因此所求问题最终}$$

转化为, 是否只存在一个  $x(0 \leq x \leq n)$ , 使得下式成立。

$$y = \sum_{i=x+1}^n \ln i - \sum_{p=1}^{n-x} \ln p - \ln(m-1)x = 0 \quad (4-7)$$

由于该函数是离散的, 所以我们利用 Matlab 来仿真不同  $m$  和  $n$  取值情况下,  $y$  函数的函数曲线, 看是否只存在一个与  $x$  轴的交点。

经过几十组的模拟, 发现无论  $m$  和  $n$  如何取值,  $y$  函数曲线和  $x$  轴始终只有一个交点, 如图 4.4~4.7 所示为其中四组情况得到的曲线。也就是说, 式 (4-1) 只存在一个最大值, 即对于一个时隙而言, 只有在一个特定  $x$  下, 它被  $x$  个 CM 同时选中的概率是最大的。

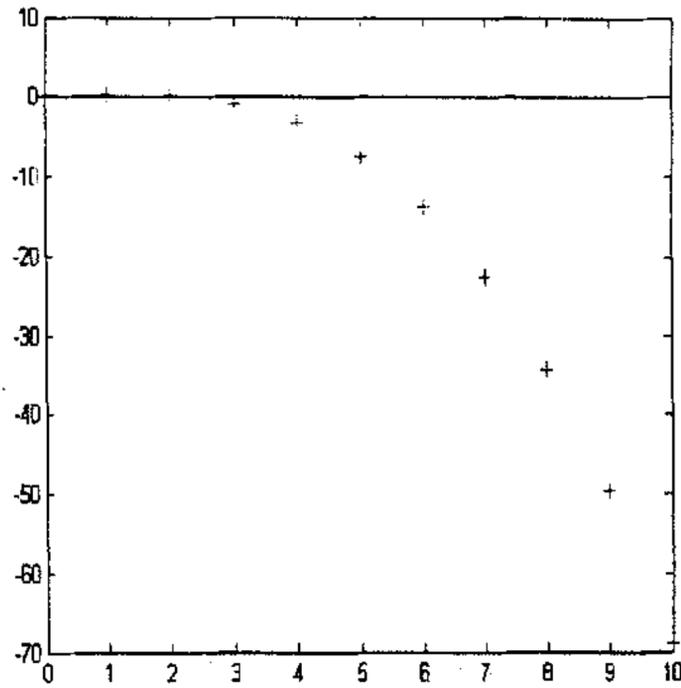


图4.4  $n=10, m=8$

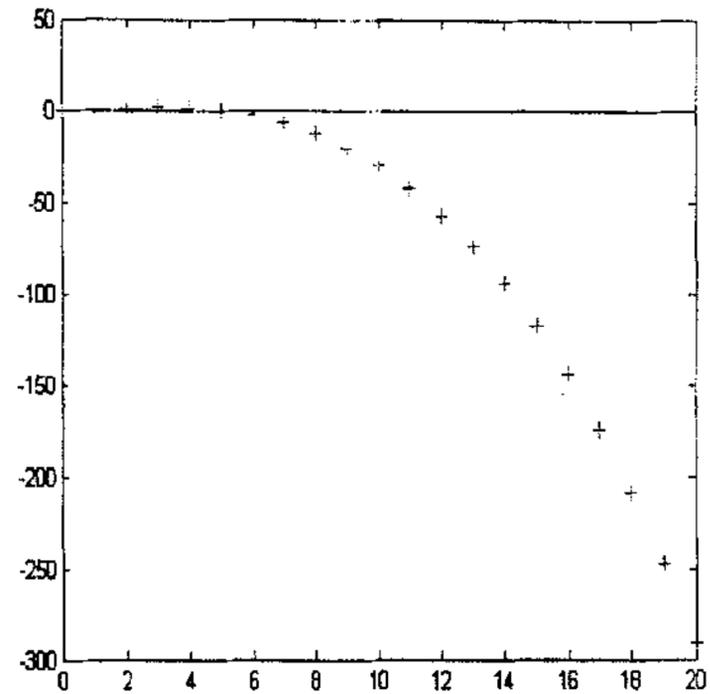


图4.5  $n=20, m=10$

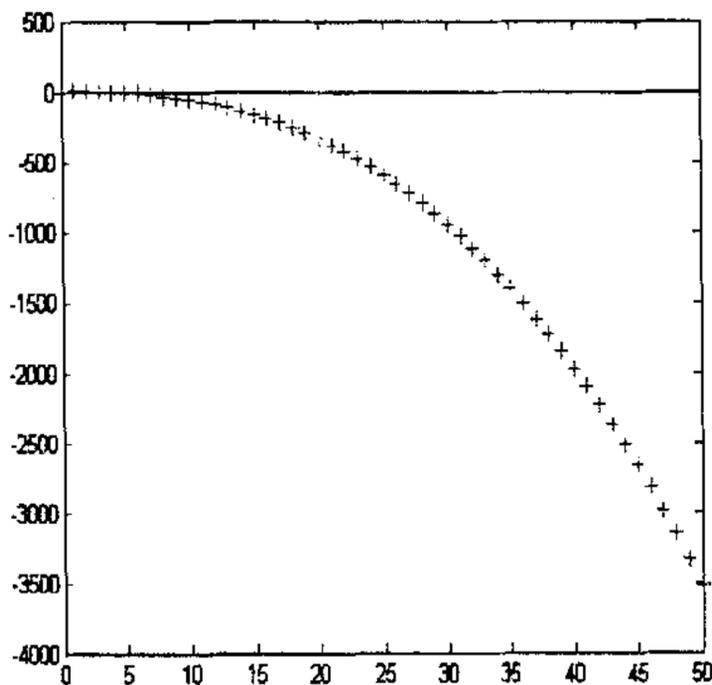


图4.6  $n=50, m=40$

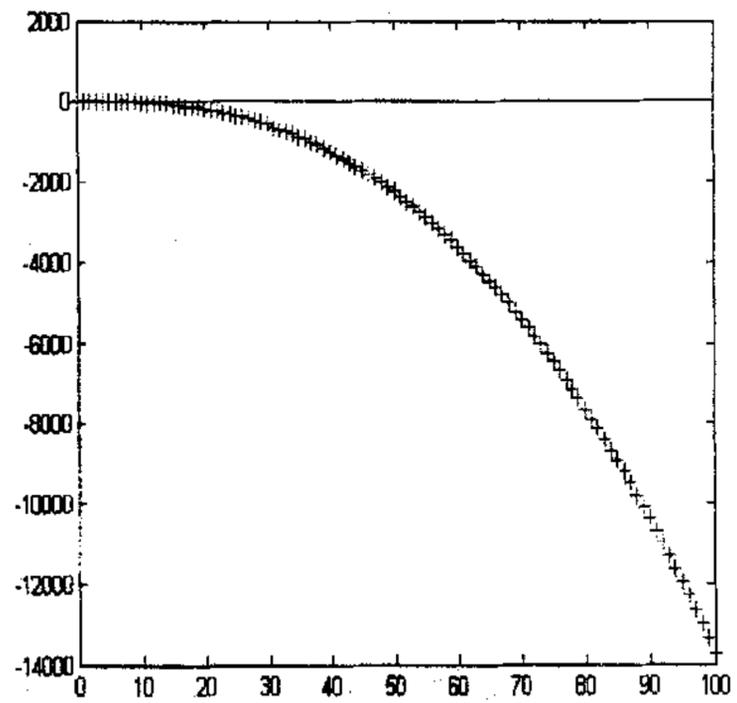


图4.7  $n=100, m=40$

由上面的分析, 可以得出无论 $m$ 和 $n$ 如何取值,  $p(x)$ 的变化曲线都始终是一个先增加后减小的曲线。当 $x$ 大于或等于2时, 总有一个并且只有一个 $x$ 的取值使得概率最大。也就是说, 当时隙发生冲突时, 总有一个并且只有一个最大可能的冲突用户 (CM) 数目。那么这个数目一定是2吗? 并且其他数目发生的概率有多大呢? 我们使用Matlab来模拟式 (4-1) 在不同情况下的函数曲线。

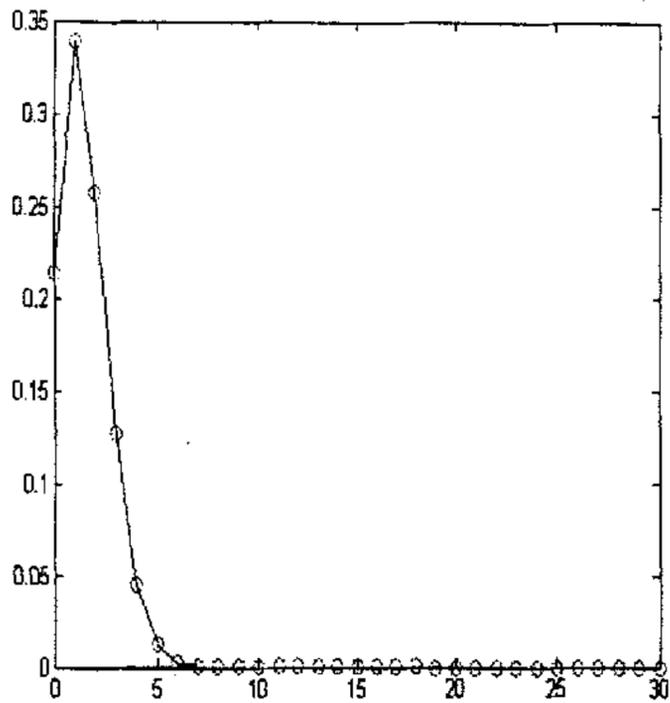


图4.8  $n=30, m=20$

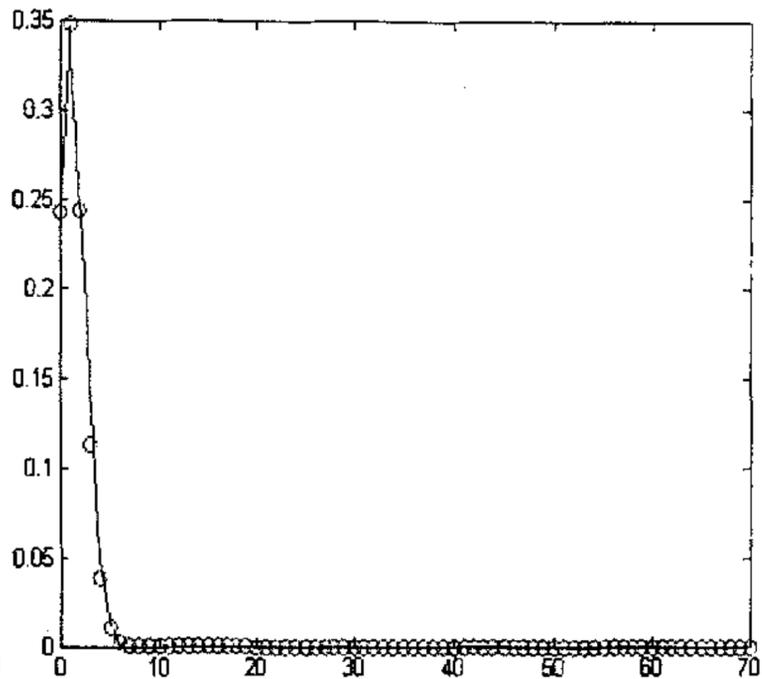


图4.9  $n=70, m=50$

由图4.8和4.9，我们发现几点。首先，曲线确实是先增后减。其次，在当前 $n$ 和 $m$ 相差不太大的情况下，时隙被一个用户选中的概率是最大的，即竞争成功的概率最大。在发生冲突的情况下，时隙被两个用户同时选中的概率是最大的，但是同时被三个用户选中的概率约为两个用户的50%。第三，曲线比较陡峭，当 $x$ 大于一定数值后，概率就近似为0了。在这两组中， $x$ 都是大约从6以后，生成的概率几乎为0。

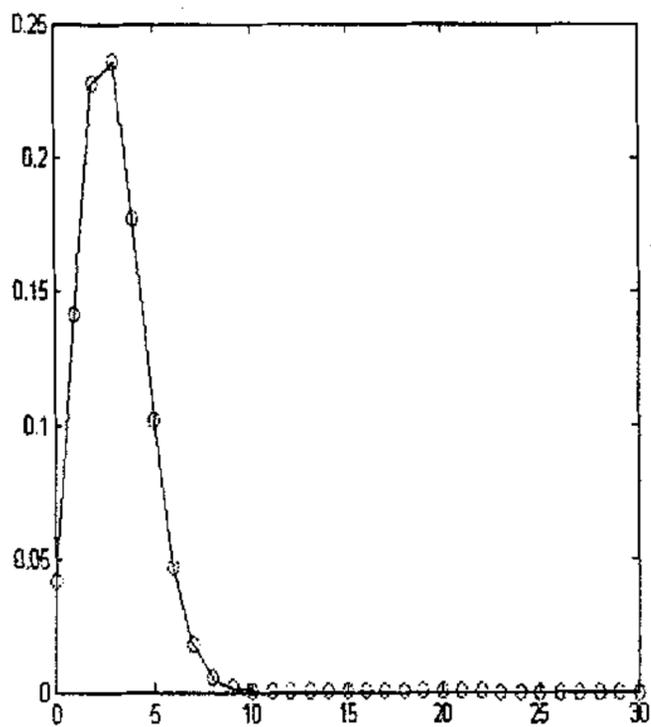


图4.10  $n=30, m=10$

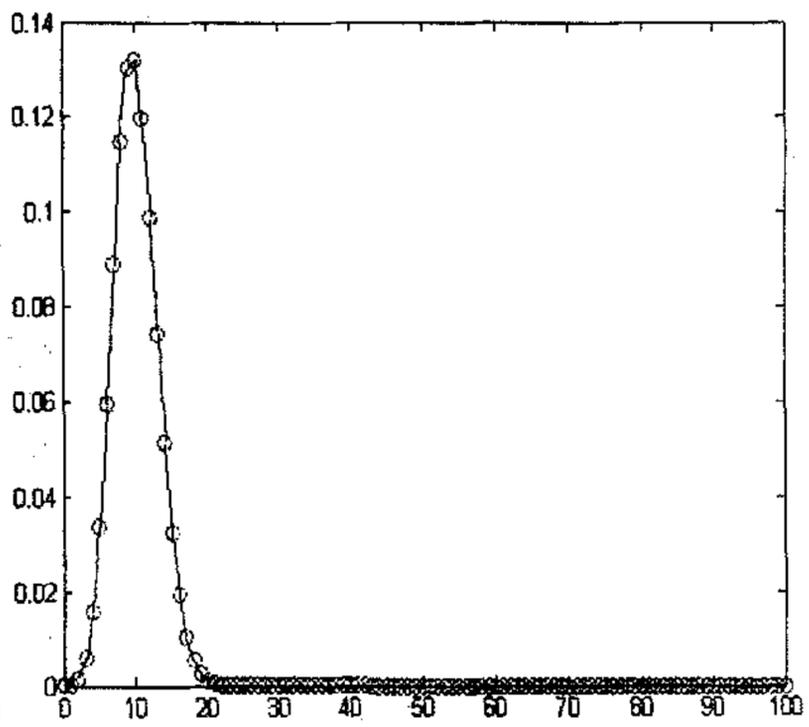


图4.11  $n=100, m=10$

由图4.10和4.11，当 $n$ 和 $m$ 的差距增大以后，即可使用的竞争时隙个数远小于用户个数的时候，情况就与前面两组发生了变化。虽然曲线依然是一个比较陡峭的先增后减，而且当 $x$ 大于一定的数值后概率也近似为0，但是最大概率并

不是出现在 $x=1$ 。当 $n=30$ ,  $m=10$ 时,  $x=3$ 时, 概率最大, 并且其他冲突情况的概率也相差不大, 如 $x=2$ 时概率与 $x=3$ 时非常接近。而图4.11, 是我们所取的一种极限情况, 即 $n=100, m=10$ 时。我们发现甚至要到 $x=11$ 时, 概率才能达到最大, 而且各种冲突情况的概率都非常接近。

综上所述, 当一个竞争时隙发生冲突时, 最大可能的冲突的CM个数是与窗口大小(即窗口中的竞争时隙个数)和用户个数密切相关的。当时隙个数比较接近用户个数的时候, 发生冲突的可能性相对比较小。这时, 一旦发生冲突, 最大的可能是两个CM同时选中了这个时隙, 但是, 也有一定可能(概率随情况而变化)是三个用户同时选中。当时隙个数和用户个数相差比较大时, 用户只能在比较少的竞争时隙中进行随机选择, 所以冲突的概率比较大。一旦发生冲突, 最大可能是多个CM同时选中这个时隙, 而且随着回退窗口大小和用户个数差距的增大, 最大可能冲突CM个数也在增大。在初步掌握了如何获知冲突CM的个数后, 就可以利用其来解决冲突竞争。

#### 4.4.2. 竞争解决

如何调整竞争时隙的个数, 即如何调整窗口的大小, 这是HFC宽带接入网设计中的关键问题。由于DOCSIS协议并没有对这一问题做出具体规定而是留给了开发者, 所以研究者们提出了许多的方案。不管用什么办法, 其中心思路都是要做到使竞争时隙尽可能被成功使用, 而且尽可能地减少冲突的可能性。

由式(4-1)还可以得到一个有用的结论。当 $x=1$ 时,  $p(x)$ 表示只有一个用户选中这个时隙, 即该时隙被成功竞争利用的概率。现在, 设 $m$ 为自变量,  $n$ 为已知的用户个数, 则式(4-1)可以化成:

$$p(m) = n \frac{1}{m} \left(1 - \frac{1}{m}\right)^{n-1} \quad (4-8)$$

对式(4-8)求导:

$$p(m)' = \left[ n \frac{1}{m} \left(1 - \frac{1}{m}\right)^{n-1} \right]'$$

$$\begin{aligned}
 &= n \left[ \left( \frac{1}{m} \right)' \left( 1 - \frac{1}{m} \right)^{n-1} + \frac{1}{m} \left( \left( 1 - \frac{1}{m} \right)^{n-1} \right)' \right] \\
 &= n(n-m) \left( 1 - \frac{1}{m} \right)^{n-2} \frac{1}{m^3} \tag{4-9}
 \end{aligned}$$

为了求  $p(m)$  的最大值, 所以令式 (4-9) 等于0, 得到:

$$p(m)' = n(n-m) \left( 1 - \frac{1}{m} \right)^{n-2} \frac{1}{m^3} = 0$$

$\therefore m = n$  或  $m = 1$  (舍去)

$m = 1$ , 即竞争时隙个数只有一个, 不是概率的最大值。所以  $m = n$ , 即当窗口大小 (竞争时隙个数) 和用户个数相等时, 时隙被成功利用的概率最大。所以, 只要能得到用户个数, 调整竞争时隙个数使之等于用户个数, 就可以使时隙被最大可能地利用成功。这也是本竞争解决算法的基本思路, 见图4.12。

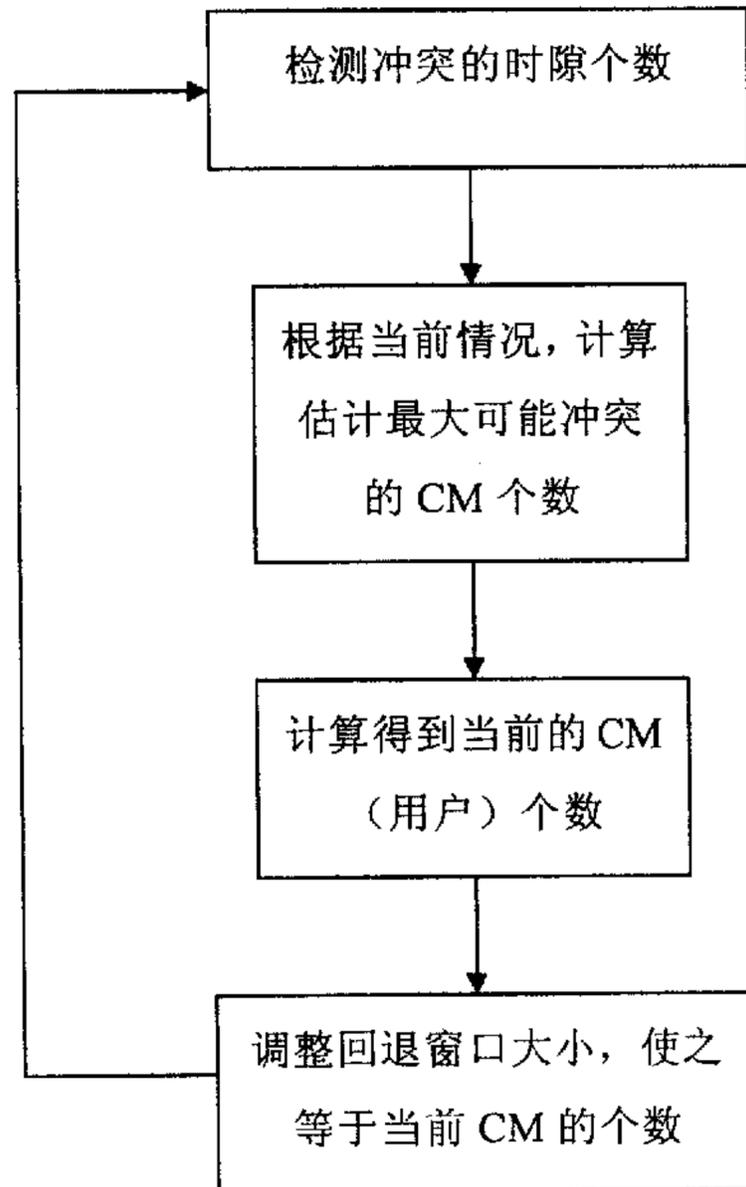


图4.12 竞争解决算法的基本思路

#### 4.4.2.1. 平均最大可能冲突 CM 数

由4.4.1的分析, 虽然最大可能冲突CM数目存在, 但是它的概率并不是远大于其他数目的概率, 所以不能完全忽略其他数目。在这里, 我们用数学期望  $E(x)$  来计算最大可能冲突CM的数目, 这样就能很好地表示出平均情况下, 最大可能冲突的CM的个数。

观察图4.8~4.9, 由于概率函数曲线比较陡峭, 所以这就给我们的计算数学期望带来了好处。首先,  $x$ 大于一定值后, 函数值就近似为0, 可以不予考虑。在图4.9中, 即便是 $n=100$ , 在 $x$ 大于等于18以后, 都近似为0了。其次, 对于小于最大概率的30%的情况可以不予考虑。当然, 这个百分比可以作为一个参数进行调整: 百分比越小算出的数学期望就越准确但是计算量就越大。最后, 函数值在曲线两侧的分布是比较均匀的, 所以可以估计这样算出的数学期望值与最大概率的那个CM数目应该还是比较接近的。

所以, 计算最大可能冲突数目的数学期望的思路是:

- (1) 在函数曲线上找到概率最大的点。这个点就是最大可能冲突CM的数目。
- (2) 以这个概率的30%为界限, 统计大于这个界限值的点 ( $x=0$ 和 $x=1$ 不计入)。这些点就是不能忽略的可能冲突的CM数目。
- (3) 按照这些点在冲突情况中各自所占的比例计算他们的数学期望值。

我们用4.4.1中的数据为例, 当 $n=5$ ,  $m=3$ 时:

X	0	1	2	3	4	5
$p(x)$	13.169%	32.922%	32.922%	16.461%	4.115%	0.412%

首先, 最大可能情况是 $x=2$ 。然后, 大于它的概率的30%的点是 $x=3$ 。最后, 按照这两个点的概率值的比例计算数学期望  $E(x) = 2.3$ 。这个值就是在 $n=5$ ,  $m=3$ 的情况下, 当一个时隙发生冲突时, 平均最大可能冲突的CM的个数。

由前面的分析, 我们的调整是要使窗口中竞争时隙个数和用户个数相等, 所以下面就分析这种特定情况, 当 $n=m$ 时的数学期望。

设 $n=m$ , 式(4-1)化为:

$$p(x) = C_n^x \left(\frac{1}{n}\right)^x \left(1 - \frac{1}{n}\right)^{n-x} = \frac{n!(n-1)^{n-x}}{x!(n-x)!n^n} \quad (4-10)$$

设数目为 $n$ 时, 概率为 $p_1(x) = p(x)$ ; 数目为 $n+1$ 时, 概率为 $p_2(x)$ , 则:

$$p_2(x) = \frac{(n+1)!n^{n+1-x}}{x!(n+1-x)!(n+1)^{n+1}}$$

为了分析 $p_1$ 和 $p_2(x)$ 的关系, 将两者相比, 得到:

$$\begin{aligned} \frac{p_1(x)}{p_2(x)} &= \frac{\frac{n!(n-1)^{n-x}}{x!(n-x)!n^n}}{\frac{(n+1)!n^{n+1-x}}{x!(n+1-x)!(n+1)^{n+1}}} \\ &= \frac{n!(n-1)^{n-x}}{x!(n-x)!n^n} \cdot \frac{x!(n+1-x)!(n+1)^{n+1}}{(n+1)!n^{n+1-x}} \\ &= \frac{(n+1)-x}{n+1} \left(\frac{n-1}{n}\right)^{n-x} \left(\frac{n+1}{n}\right)^{n+1} \end{aligned} \quad (4-11)$$

观察式(4-11), 会发现当 $n$ 和 $1$ 相差比较大时,  $n \approx n+1 \approx n-1$ , 又当 $x$ 与 $1$ 相差很小时,  $n-x \approx n$ , 则式(4-11)化为:

$$\frac{p_1(x)}{p_2(x)} \approx \frac{n-x}{n} \approx 1 \quad (4-12)$$

所以, 当用户个数和 $1$ 相差比较大时, 不同的用户个数在 $x$ 比较小的时候得到概率值十分接近。

表4.1是在不同 $n$ 取值情况下实际计算所得的概率值:

表4.1 不同情况下的概率

	X=1	2	3	4
n=5	0.4096	0.2048	0.0512	0.0064
8	0.3927	0.1963	0.0561	0.0100
10	0.3874	0.1937	0.0574	0.0112

30	0.3741	0.1871	0.0602	0.0140
80	0.3702	0.1851	0.0609	0.0148

分析上面数据, 当n小于10时, 满足前面计算数学期望的x取值只有2。当n大于10以后, x=1~4所得到的概率值都非常接近, 并且x=3的概率值都为x=2的1/3。按照该比例计算数学期望值为2.25。

因此, 我们可以得到如下一个非常有价值的结论: 当窗口中竞争时隙个数和用户个数相等时, 平均最大可能冲突CM的数目可以看作是一个固定值, 并不会随着用户个数的变化而变化。当用户数目小于10时, 数学期望值为2; 当用户数目大于或等于10时, 数学期望值为2.25。

#### 4.4.2.2. 算法流程

基于上面各章节的分析, 现在我们就能够得到当前发出请求申请的用户个数。首先, CMTS检测上行信道, 检测出竞争成功的时隙个数和冲突的时隙个数, 然后按照如下算法计算发出请求的用户个数:

$$\text{发出请求的用户个数} = \text{竞争成功的时隙个数} + \text{冲突的时隙个数} \times \text{平均最大可能冲突CM数目} \quad (4-13)$$

由公式(4-1), 计算平均最大可能冲突CM数目, 需要知道当前的竞争时隙个数和用户个数。竞争时隙个数是由CMTS每次都分配好后预先通知CM的, 是已知量。而对于用户个数, 我们把它分为先后两个部分: “初始用户数”和“检测用户数”, 如表4.2所示。

表4.2 参数设置

第I次发出申请的用户个数	$n_i$
第i次分配的竞争时隙个数	$m_i$
第i次的初始用户数	$n_{i0}$
第i次的检测用户数	$n_i$
第i次竞争成功的时隙个数	$m_{iS}$

第 <i>i</i> 次冲突的时隙个数	$m_{iF}$
---------------------	----------

实际的请求申请, 是一个比较平稳的过程。在短时间内, 发出申请的用户个数的增加量和减小量都非常小, 也就是说, 发出请求的用户个数的变化是比较平缓的, 不会有太大的突然增加和突然减小。在极短的时间内, 可以认为用户个数近似相等。所以, 我们先设第*i*次的发出申请的用户个数为“初始用户数”, 就等于第*i*-1次实际发出申请的用户个数, 即:

$$n_i = n_{i_0} = n_{i-1} \quad (4-14)$$

用当前这个  $n_i$  来计算第*i*次的平均最大可能冲突的CM个数。所得的值便是在当前的  $n_i$  和  $m_i$  的条件下, 得到的第*i*次的数学期望值。

然后根据检测上行信道的结果, 由式 (4-13) 得到“检测用户数”, 再设第*i*次发出申请的用户个数为“检测用户数”。这也是第*i*次竞争中, 实际发出申请的用户个数。

$$n_i = n_{i_1} = m_{iS} + m_{iF} \times E_i(x) \quad (4-15)$$

最后, CMTS将第*i*+1次的竞争时隙个数设置成第*i*次的发出申请的用户个数, 作为第*i*+1次的窗口大小写入MAP帧中发往CM。

这样, CMTS就完成了一次竞争解决的窗口调整。需要注意几个关键点:

(1) 根据4.4.2.1中的分析, 只要竞争时隙个数和用户个数相等, 我们就不要再计算数学期望值, 而直接代入前面的结论。所以, 只有当第*i*次CMTS调整窗口的时候, 无法在一个MAP帧中分配足够多的竞争时隙去等于用户个数的时候, 即窗口小于用户数, 则在第*i*+1次中, 才需要去计算第*i*+1次的数学期望值。

(2) 为了防止某一次偏差太大的检测值造成的误差, 我们检测连续的三组竞争时隙, 然后计算这三组的平均成功时隙个数和平均冲突时隙个数, 才进行一次调整。也就是说, 每三个MAP帧为一个循环, 每三个才重新调整一次, 作为一个新的MAP帧发往CM, 而接下来连续两个MAP帧和这一个的竞争时隙分配相同, 以此循环进行。

算法的流程如图4.12所示:

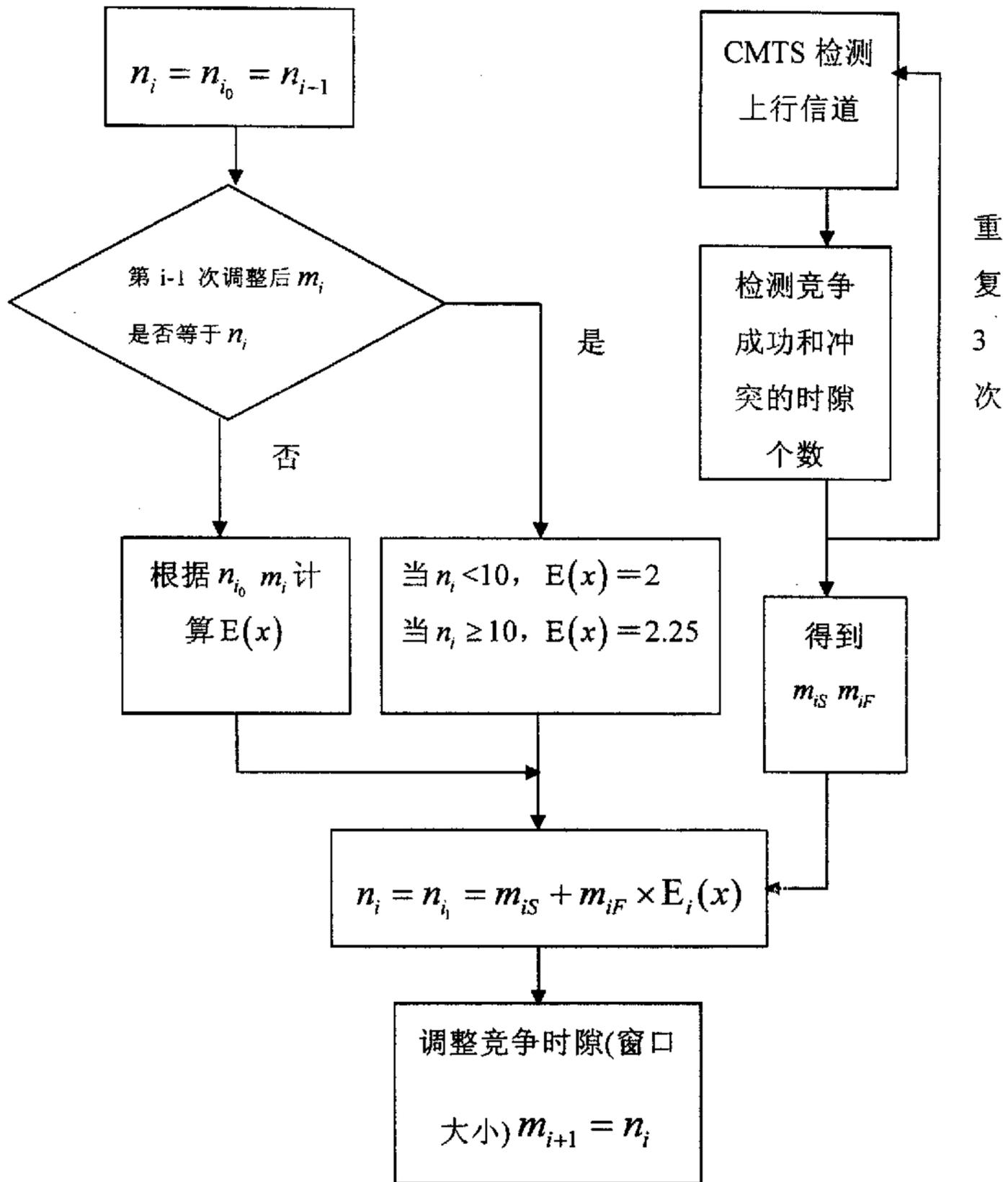


图4.12 竞争解决算法流程图

下面将基于冲突检测的竞争解决算法用伪代码的形式描述:

```

n_{i_0} = n_{i-1};
n_i = n_{i_0};
if (m_i = n_{i_0})
    { if (n_{i_0} < 10) E_i(x) = 2;
      else E_i(x) = 2.25;
    }
  
```

```

else compute_E( $n_{i_0}$ ,  $m_i$ );

 $n_{i_1} = m_{iS} + m_{iF} * E_i(x)$ ;

 $n_i = n_{i_1}$ ;

 $m_{i+1} = n_i$ ;

```

其中函数 $\text{compute\_E}(n_{i_0}, m_i)$ 为计算具体数学期望的函数。首先从 $x=0$ 开始计算式(4-1)的函数值。根据前面得到的式(4-1)的函数曲线先增后减, 函数只有一个最大值, 所以计算得到的函数值开始是从小到大。一旦发现新得到的函数值比前一个小, 则说明前面一个即为最大值。然后, 计算得到的函数值又开始从大到小, 一旦新计算的函数值小于最大值的30%, 则停止计算。已经计算的点即为求数学期望所必须的点。

#### 4.5. 讨论和仿真

根据第三章对DOCSIS协议的研究分析, CMTS是先将时隙分配好, 再用MAP帧通知CM。所以, CMTS对于时隙的调整不可避免地存在一定的滞后性。因此, 竞争解决算法可以形象地说是先观察, 再调整, 再观察。它是一个不断适应调整的过程, 不断地使时隙的分配能够适应新观察到的情况。

当实际发出申请的用户个数增加, 由于时隙分配是在之前做出的, 所以经过多次对竞争的观察, 必然发现冲突增多, 也就是导致 $m_{iF}$ 的增加。由式(4-15), 由于存在数学期望这个倍数关系, 所以必然导致“检测用户数”增大, 大于了“初始用户数”。这样, CMTS就会相应地增加竞争时隙, 也就是增大窗口。同理, 当实际发出申请的用户个数减少, 时隙个数相对于用户来说比较宽余, 就导致冲突减少, 从而造成“检测用户数”减小, 小于了“初始用户数”。这时, CMTS就会相应地减小窗口。因此, 通过式(4-15)数学关系的控制, 达到了根据实际情况控制窗口扩大和缩小的目的。

需要注意的是我们可以从以下的途径来降低本算法的运算量, 从而降低算法的时间复杂度。首先, 在网络运作比较平稳时, 大多数情况下都不需要做计算而直接代入固定的数学期望值。其次, 当需要计算数学期望值时, 在寻找最大冲突用户数时, 实际需要计算的点并不多。根据前面Matlab仿真结果, 即便是在用户数为100, 竞争时隙数为10的极端情况下也仅仅需要计算15个点就可以

找到结果。第三, 对于计算量最大的阶乘, 可以根据第五章的算法, 划分优先级。由于每一个优先级内只是一部分用户, 用户数目减小, 因此相对于计算全部用户, 阶乘的计算量大大减小了。

#### 4.5.1. 仿真环境介绍

我们所采用的仿真环境是美国MIL3公司的网络仿真软件OPNET Modeler。OPNET Modeler是一种商用化的网络仿真软件, 在全球范围有很多著名网络通信厂商使用OPNET Modeler, 包括3Com、CableLabs、Cisco、LSI、Microsoft、Motorola、Nortel、Philips、TI、AT&T 等<sup>[14]</sup>。其中CableLabs正是DOCSIS协议的制定者。OPNET Modeler支持多种协议的建模和仿真, 而且在它上面实现新的协议也相当灵活和容易。首先, OPNET Modeler 有友好的用户图形界面, 研究人员可以很容易的利用鼠标拖动所需的模块建立自己的模型。其次, OPNET Modeler的体系结构非常明确, 把整个网络体系分成三层: 节点、模块和过程。节点由模块组成, 而每个模块又有自己的有限状态机控制模块的运行, 有限状态机中的过程又是由一种类似C语言的Proto C语言实现, 相对灵活而且仿真的速度快。

#### 4.5.2. 仿真结果及讨论

我们将该算法和固定窗口大小的二进制指数回退算法做计算机仿真, 比较这两种算法的接入延迟。

固定窗口大小即每次MAP帧分配的回退窗口的竞争时隙数目都是相同的。我们设一共有32个用户, 发出申请的用户数目服从均值为10的泊松分布。每次用户申请的数据量分成两种情况: 64byte和800byte。表4.3列出主要参数。

表4.3 仿真参数设置

上行数据速率 (UPSTREAM_DATA_RATE)	5.12Mbps
CM数量	32个
微时隙尺寸 (mini-slot size)	16byte
数据包大小 (packet size)	仿真一: 64byte

	仿真二: 800byte
时间/微时隙 (second per mini-slot)	25 $\mu$ s
微时隙/秒	40,000个
仿真时间 (MAX_SIMTIME)	1,200,000个微时隙, 约30秒
Ranging Backoff Start	3
Ranging Backoff End	6

这两种算法的网络负载和接入延迟关系曲线如图4.13和图4.14所示。分析仿真结果, 我们可以发现, 在这两种数据包大小的情况下, 基于冲突检测的竞争解决算法的接入延迟都比固定窗口大小的算法要小。这是因为本算法通过对回退窗口的竞争时隙数目的调整, 使得竞争时隙被充分利用, 而减少了冲突。在网络负载很小的时候, 该算法的优势还不明显, 这是因为此时发送申请的用户数目很小, 所以冲突很少。但是, 当网络负载增加以后, 随着用户数目和冲突地加剧, 本算法的优势就逐步体现了出来, 明显比固定窗口大小算法的延迟要小。比较图4.13和图4.14, 我们发现在大数据包情况下, 本算法的优势领先不如在小数据包的情况下。这是因为, 大数据包, 特别是在高网络负载条件下, 大量的时隙都被分配用于传输数据, 竞争时隙大量减少, 而使本算法无法完全按照要求调整窗口的大小。

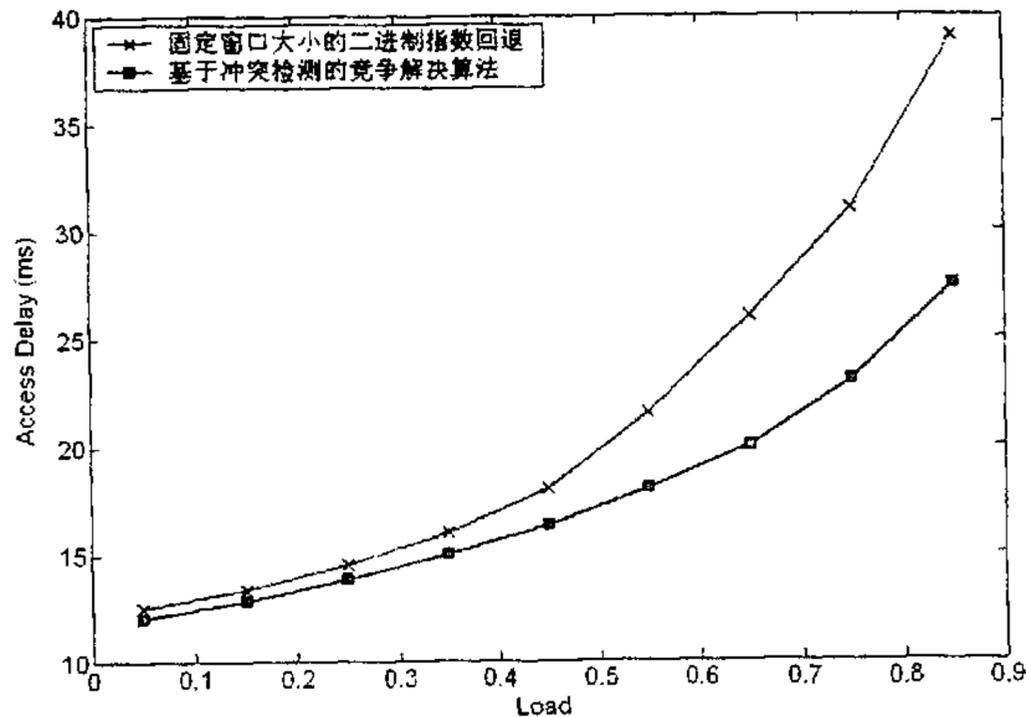


图4.13 两种算法的关系曲线 (数据包大小为64byte)

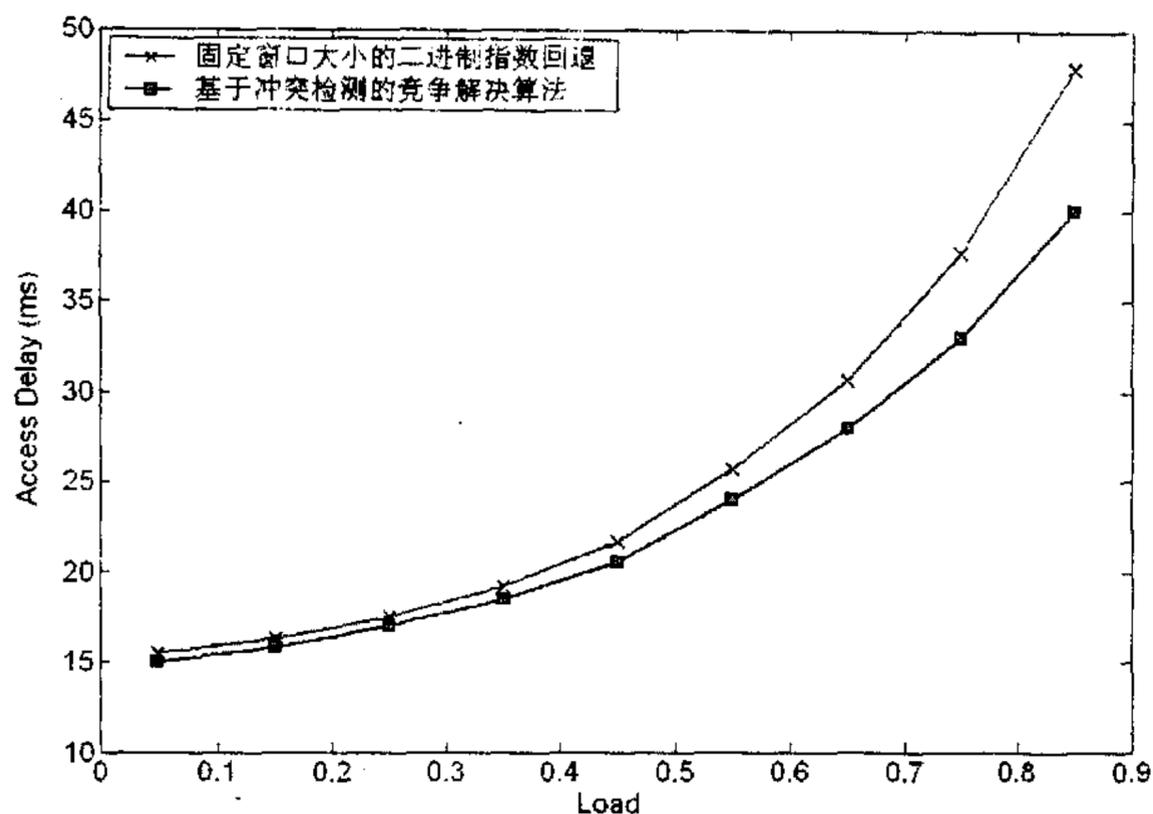


图4.14 两种算法的关系曲线 (数据包大小为800byte)

#### 4.6. 小结

本章分析了HFC网络存在的三个问题，双向改造困难、噪声和无法检测冲突。根据这三个问题，提出了用无线传输方式来代替HFC网络的上行信道，从而使CMTS具有了检测信道的能力。然后利用CMTS可以检测冲突的改进，设计了基于冲突检测的竞争解决算法。本算法可以根据冲突的剧烈程度，调整回退窗口的大小，有效地减少冲突，以适应用户数目的不断变化。

## 第5章 基于优先级的冲突解决算法研究

### 5.1. 研究问题分析

在上一章中，我们设计了基于冲突检测的竞争解决算法。在这个算法中，当发出申请的用户个数增加的时候，要增加竞争时隙的个数。同时，我们也提出了可能存在的一种情况，即CMTS调整窗口的时候，无法在一个MAP帧中分配足够多的竞争时隙去匹配用户个数。这在HFC宽带接入网的运作中是一个常见的问题。<sup>[15]</sup>从理想状态来讲，在总的时隙个数大致一定的情况下，竞争时隙和数据时隙都是希望尽可能多的被分配。但是，就实际而言，显然不可能都同时满足。而且由于上行信道的主要任务是传输CM的上行数据，所以往往会造成竞争时隙的个数总是不能满足分配的需要。

这种情况将导致一个重要问题：延时。在前面章节中，为了讨论的方便，我们都假设每一次MAP帧中分配的竞争时隙个数就是该次设置的回退窗口大小。但是如果一旦在一个MAP帧中可分配的竞争时隙个数小于该次设置的回退窗口大小，会出现什么后果呢？我们分两种情况进行讨论。

(1) 当CMTS发现当前没有足够多的时隙能够被分配成竞争时隙去满足需要，它仍然可以将回退窗口大小按照需要去设置，如按照上一章所设计的竞争解决算法。但是，这就意味着会有一部分的窗口范围落在了下一个MAP帧中。对于CM而言，它只能随机地在这个已经规定好的窗口范围内随机地选择，所以很可能CM选中的回退大小值就落在了下一个MAP帧中，如图5.1所示。

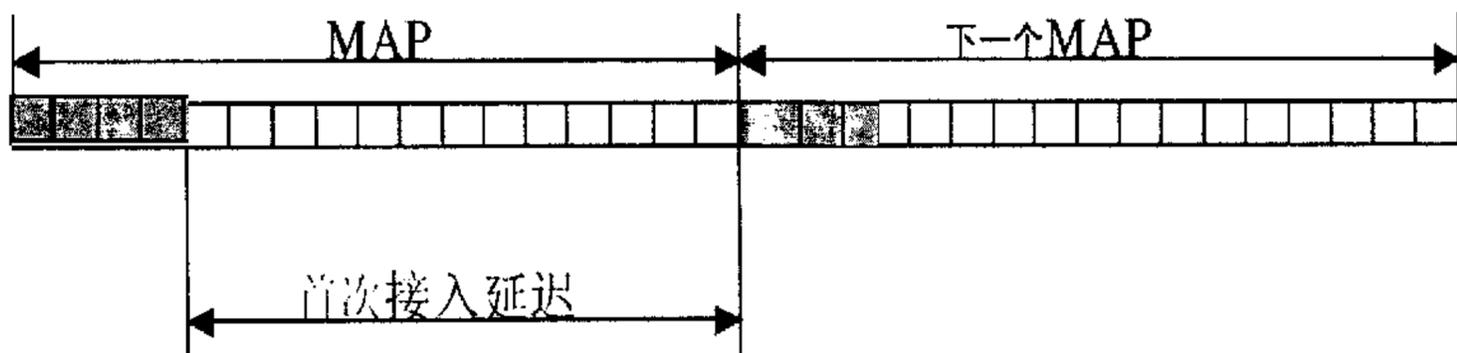


图5.1 窗口与MAP关系示意图

上图中黑色部分即为竞争时隙，可以看到，当前MAP中只能分配4个竞争时隙，而窗口大小为7，所以就有3个属于窗口的竞争时隙是在下一个MAP帧中

才进行分配。如果CM随机选择的时候,选中了后面3个,虽然该CM是从当前MAP规定时间开始竞争,但是也必须等到下一个MAP的时间才能发送请求信息。这样,产生的接入延迟就如图上所示。与窗口在一个MAP帧中的情况相比较,延迟时间增加了,增加的长度为当前MAP帧中分配的数据时隙(白色部分)的时间长度。而且如果当前上行数据增多,导致数据时隙分配增多的时候,那么延迟还会加大。这还是没有考虑冲突的延迟,如果一旦发生冲突,延迟时间就还要加上第二次发出申请的延迟时间。

(2) 当CMTS发现当前没有足够多的时隙能够被分配成竞争时隙去满足需要,它可以重新设置窗口大小,也就是说,当前MAP帧中有多少竞争时隙可使用,就将窗口设置成这个大小。这样就可以避免上面一种情况所出现的跨MAP帧的接入延迟。但是这样会导致窗口不能满足需要,必然导致在该窗口内冲突地增加。总体上来讲,由于更多的CM要发出第二次申请,所以平均的延迟时间会有一定地增加。但是,由于HFC宽带接入网的上行信道的带宽资源比较宝贵,所以大部分时候,竞争时隙的分配是处于一种不能足够满足需要的状态下。那么(1)(2)两种情况下,都会有冲突的延迟,但是又由于第一种情况还多出了一个跨MAP帧的首次接入延迟,所以相对来讲,第二种情况的平均延时要小一些。我们现在先不考虑冲突后的延迟,只是从首先接入的延迟时间上来看,窗口大小范围落在一个MAP帧中的时候必然比落在两个或者多个MAP帧中要小地多。

综上所述,既要尽可能地按照发出申请的用户需要去设置窗口大小,又要尽量地减少首次接入的延迟,所以,我们引入了优先级的概念。

## 5.2. 基于优先级的冲突解决算法

### 5.2.1. 背景分析

DOCSIS协议中并没有规定优先级,也没有引入优先级的概念,这是为研究者留下足够多的空间去设计具体的算法。

在网络业务中,有些用户对于实时性要求比较高,对延迟非常敏感,比如VoIP;但也有一些用户对延迟并不太敏感,比如浏览网页。在DOCSIS协议中并没有优先级的区别,所以他们都享有公平的机会去竞争。因此,实时性要求

高的用户很可能就被实时性低的用户竞争而挤掉了, 从而造成延迟加大, 这对实时性用户来说是致命的。

同时, 对于网络服务商来说, 他们有必要对用户群做出划分。比如按照交纳的服务费划分成不同等级的客户; 或者按照用户所处位置, 某些位置上的用户是重要客户 (VIP)。

可见从实际的应用上, 划分优先级有其必要性, 更重要的是, 划分优先级能够解决我们在上一节提出的两个问题。注意, 如何划分优先级不是该算法要考虑的问题, 留待其他研究去确定, 我们专注的是已经划分好优先级以后, 网络如何按照该优先级去运作, 特此说明。

### 5.2.2. 算法原理

根据第一节的分析, 要尽可能地按照发出申请的用户需要去设置窗口大小, 那么当用户增加的时候, 窗口大小必然增加。同时, 又要尽量地减少首次接入的延迟, 也就是说要让窗口被包含在一个MAP帧中。所以, 当一个MAP帧中的竞争时隙数目小于窗口大小的时候, 就产生了矛盾。

由于窗口大小是随用户个数变化的, 用户个数减少, 窗口大小也能减少, 所以是否能先满足一部分用户的需要呢? 由于只是一部分用户, 所以专门为他们设计的窗口大小就会比原来为所有用户设计的窗口要小, 也许就能够被包含在一个MAP帧中了。于是, 对于这部分用户而言, 既满足了他们的需要, 又减少了首次接入的延迟, 矛盾就得到了很好的化解。

因此, 我们引入优先级, 优先满足高优先级用户的需要, 在MAP帧中优先安排他们的窗口。换句话说, 这是一种折中的方案算法, 既然我们不能解决所有用户的矛盾, 那我们就优先解决一部分用户的矛盾, 使他们的需要得到满足, 然后, 再利用剩下的资源去尽可能地满足其他用户。但是, 需要特别注意的是, 虽然有优先级, 但为了尽可能多地满足用户需要, 每个优先级的用户只能使用本优先级的竞争时隙。也就是说, 高优先级的用户不能使用低优先级用户的竞争时隙, 如图5.2所示, 这与某些网络中高优先级可利用低优先级的资源方案不同。所以, 我们引入优先级的概念, 其中心思想是在时隙的分配上按照优先级来进行先后分配。

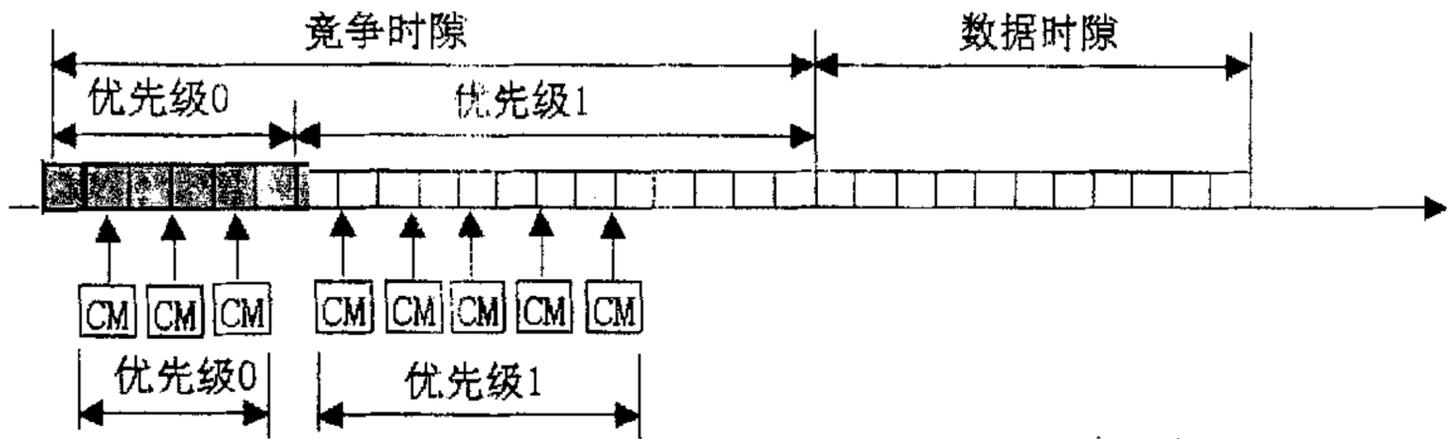


图5.2 CM按照各自优先级接入该优先级的竞争时隙

下面我们先研究最简单的情况，即只有两个优先级。

### 5.2.2.1. 两个优先级的冲突解决

假设现在已经划分好了两个优先级：优先级0和优先级1，并且优先级0高于优先级1。设它们当前这一次各自的回退窗口大小分别为： $L_0$  和  $L_1$ ，当前MAP帧中可供分配的竞争时隙的个数为 $m$ 。根据 $m$ 的多少，分成几种情况讨论：

(1) 竞争时隙个数大于或者等于这个两个窗口总大小，即  $m \geq L_0 + L_1$ ，也就是说，这两个窗口都可以被安排在这一个MAP帧中。这是最理想的状态，如图5.3所示。

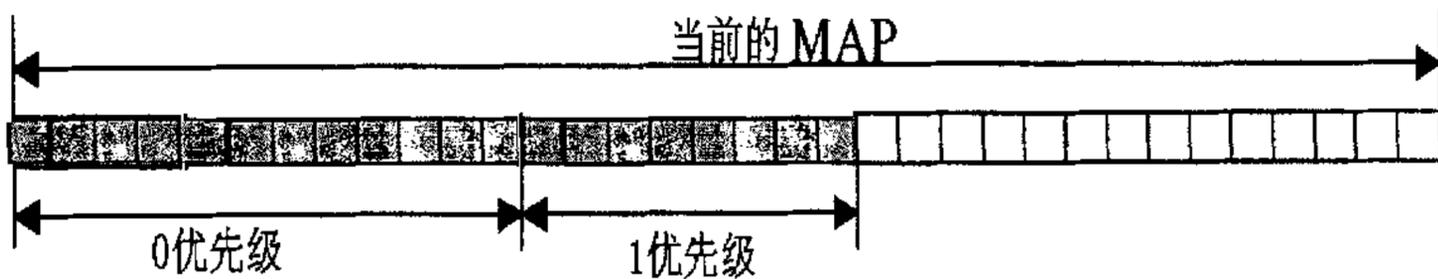


图5.3 一个MAP帧中可安排下两个窗口

如果  $m = L_0 + L_1$ ，则MAP中的竞争时隙正好可以放下两个窗口。这种情况下就不需要做任何调整。

如果  $m > L_0 + L_1$ ，也就是说，当前数据时隙不是太多，CMTS可以安排更多的时隙用于竞争。那么CMTS就可以考虑增加这两个优先级的窗口大小。在不影响其他网络运作的可允许条件下，能够设置的窗口大小肯定是越大越好。这种情况下，有两种方案可以选择。

第一，按照比例分别扩大这两个优先级的窗口；第二，将剩余的竞争时隙全部分配给优先级0。如果考虑到要满足尽可能多的用户的需要，那应该选择第

一种; 如果优先级0的用户十分重要, 应该全力保证他们的运作, 则应该选择第二种。所以, 可以根据实际情况进行方案地调整。我们这里选用第一种方案, 设调整后的两个优先级的窗口大小分别为:  $L'_0$  和  $L'_1$ , 则:

$$\begin{cases} L'_0 = m \frac{L_0}{L_0 + L_1} \\ L'_1 = m \frac{L_1}{L_0 + L_1} \end{cases}$$

(2) 竞争时隙个数小于这两个窗口总大小而大于优先级0的窗口大小, 即  $L_0 < m < L_0 + L_1$ , 也就是说优先级0的窗口大小可以被安排在这一个MAP帧中, 而优先级1则不能。在这种情况下, 优先级的制度发挥作用, 我们只保证高优先级的窗口可以尽可能地安排在一个MAP帧中。所以, 先安排优先级0的窗口大小, 剩下的时隙则作为优先级1的竞争时隙, 如图5.4所示。

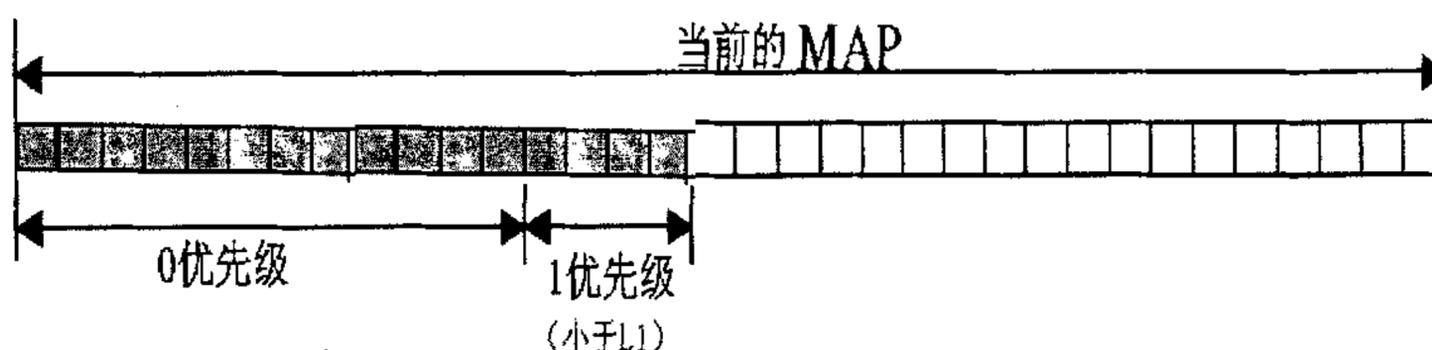


图5.4 一个MAP帧中只能安排下一个窗口

因为当前MAP帧已经无法满足优先级1本来需要的窗口大小, 只能牺牲优先级1来保障优先级0, 所以必须调整优先级1的窗口大小:

$$\begin{cases} L'_0 = L_0 \\ L'_1 = m - L_0 \end{cases}$$

(3) 竞争时隙个数小于或者等于优先级0的窗口大小, 即  $m \leq L_0$ , 也就是说, 现在连最高优先级的窗口大小都无法保证了。这样, 在优先级制度下, 我们肯定要先保证优先级0, 而不得不完全牺牲掉优先级1, 如图5.4所示。

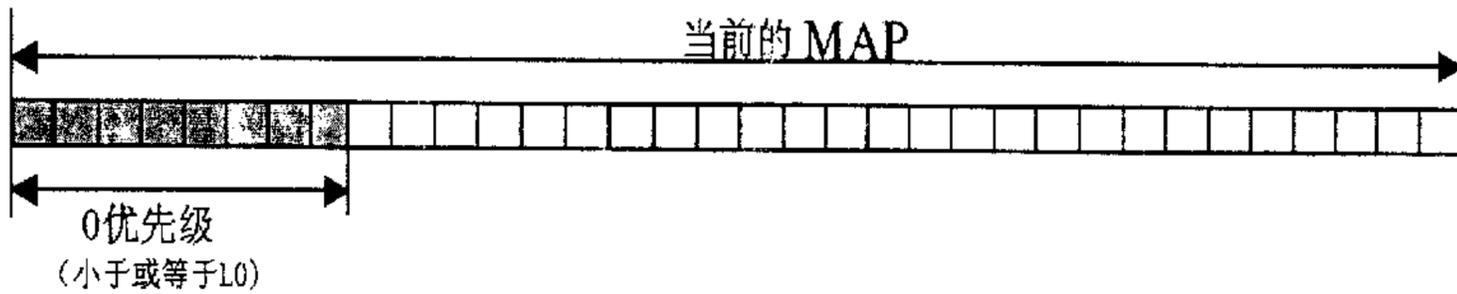


图5.5 一个MAP帧中无法安排下任一个窗口

所以，按照上述分析调整窗口大小：

$$\begin{cases} L'_0 = m \\ L'_1 = 0 \end{cases}$$

#### 5.2.2.2. 改进和流程

从上面对两个优先级的分析，我们的中心思路是要尽可能的保证高优先级能够在在一个MAP帧中设置其本来需要的窗口大小。对于低优先级，是一个最大努力服务（Best Effort）。也就是说，根据当前的情况，CMTS能够为低优先级分配就尽量分配，如果不行，就不得不牺牲低优先级用户。

在调整低优先级的窗口时，最极端的状态就如上一节所分析的第三种情况：优先级1被完全牺牲掉了。但是，在实际的网络运作中，这往往是不行。对于低优先级的用户，还是必须保证有一点最小的接入机会。比如，低优先级用户是浏览网页，那么就应该保证用户发送对网页数据的申请信息，哪怕速度非常慢。慢1秒或者慢3秒，对于这个低优先级用户来说，都不太敏感，而他们最关心的是能否打开网页，所以不应该直接就切断用户的接入。

基于以上分析，我们对上一节设计的冲突解决算法做出改进：对每一个优先级规定一个“最低竞争时隙数”。这个最低限度的竞争时隙就是为了保证用户最小的接入。

因此，在每次按照优先级进行时隙分配的时候，首先划分出每个优先级的“最低竞争时隙”。然后，再在剩下的竞争时隙中进行基于优先级的划分。而且，“最低竞争时隙数”同时也规定了MAP帧中至少必须包含的竞争时隙个数，即每个优先级的“最低竞争时隙数”的总和。

但是要注意一个问题，如果没有发生只能使用“最低竞争时隙数”的情况

时, 每个优先级里的已经提前分配好的“最低竞争时隙”也应该算入该优先级的回退窗口长度中。也就是说, 在设置窗口时, 不是去掉“最低竞争时隙数”再去分配窗口长度, 而是把它作为窗口的一部分。设“最低竞争时隙数”为 $n$ , 则5.2.2.1中的数学关系改进如下:

(1) 当  $m \geq L_0 + L_1$ , 则:

$$\begin{cases} L'_0 = m \frac{L_0}{L_0 + L_1} \\ L'_1 = m \frac{L_1}{L_0 + L_1} \end{cases}$$

(2) 当  $m < L_0 + L_1$ , 且  $m - L_0 > n$ , 即安排完优先级0的窗口后, 剩下的竞争时隙数还大于最低竞争时隙数, 则将剩下的全部分配给优先级1。

因此, 当  $L_0 + n < m < L_0 + L_1$ , 则:

$$\begin{cases} L'_0 = L_0 \\ L'_1 = m - L_0 \end{cases}$$

(3) 当  $m \leq L_0 + n$ , 即安排完优先级0的窗口后, 剩下的竞争时隙数已经小于最低竞争时隙数, 则应该保证优先级1的窗口为 $n$ 。

因此, 当  $m \leq L_0 + n$ , 则:

$$\begin{cases} L'_0 = m - n \\ L'_1 = n \end{cases}$$

(4) 由于 $m$ 最小值即为 $2n$ , 所以极限状态就是个优先级的窗口大小都是 $n$ 。这样, 就保证了每个优先级的最低接入要求。

下面就是基于优先级的冲突解决算法的流程图:

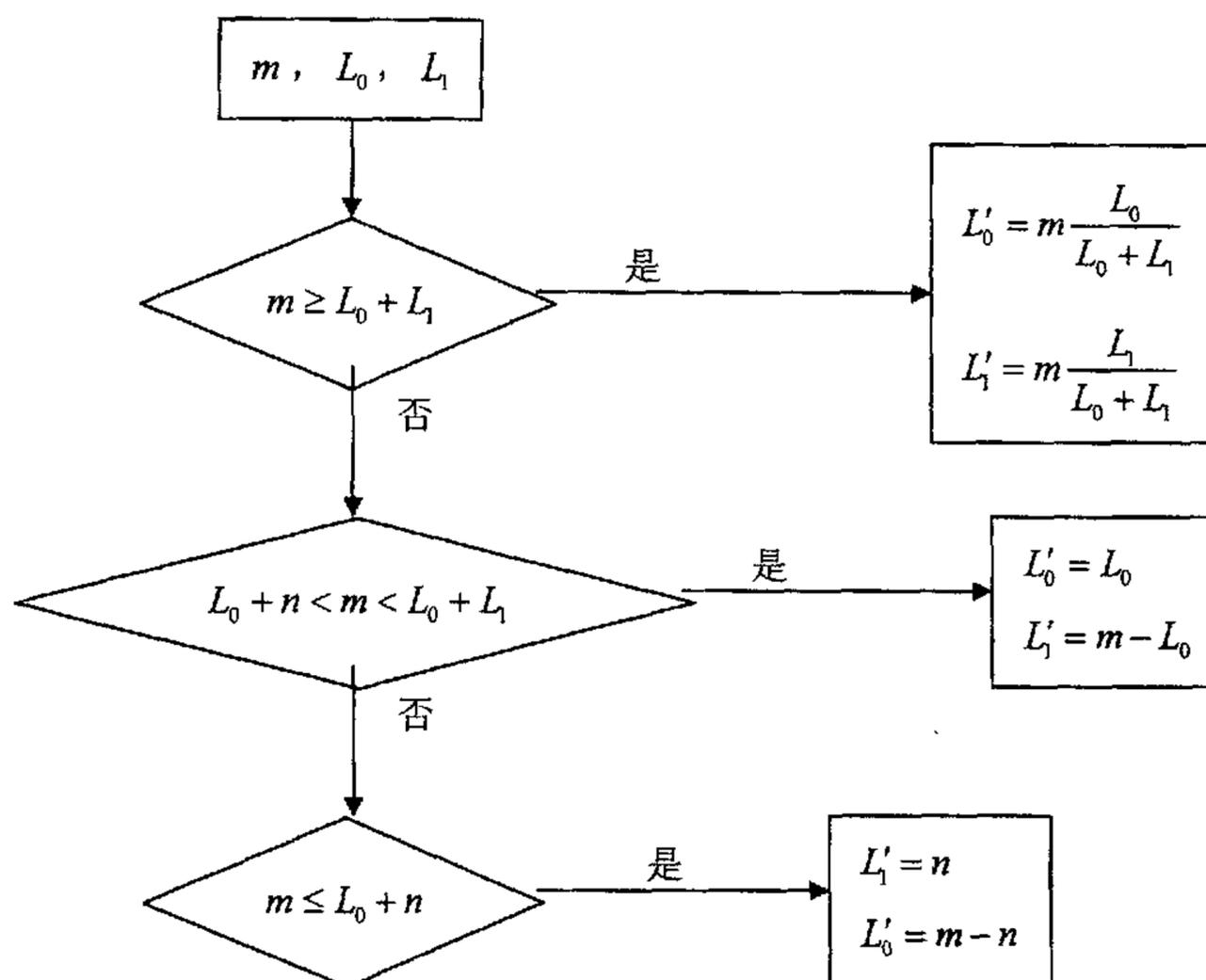


图5.6 算法流程

需要注意的是其实最后一个判断框已经不需要了，但是为了表述完整和理解的方便，还是将其写入算法流程中。

下面将基于优先级的冲突解决算法用伪代码的形式描述：

```

if (m ≥ L_0 + L_1)
{
  L'_0 = m * L_0 / (L_0 + L_1);
  L'_1 = m * L_1 / (L_0 + L_1);
}
else
{
  if (m > L_0 + n && m < L_0 + L_1)
  {
    L'_0 = L_0;
    L'_1 = m - L_0;
  }
}
  
```

```

else
{
    L'_0 = m - n;
    L'_1 = n;
}
}
    
```

### 5.3. 讨论和仿真

以上为了讨论的方便，都是基于两个优先级。当有三个和三个以上优先级的时候，思路和两个优先级一致。在流程上，根据优先级由高到低按照如图5.6所示的这个流程进行循环。

由于DOCSIS没有包含优先级，所以在它所规定的帧格式中，也没有定义优先级的选项。因此，要在MAP帧中描述对不同优先级的分配，就必须改动MAP帧。所幸的是，DOCSIS充分预留了很大的空间，使研究者能进一步开发设计。在MAP帧中，DOCSIS专门预留了“Reserved”字段，见图3.6。我们就利用这个字段，将优先级信息写入。这样，就可以定义MAP帧后面描述的时隙分配是属于哪一个优先级的。CM在获得MAP帧后，根据自己的优先级查看MAP中相应优先级的分配信息，就可以获知该优先级的窗口设置和时隙分配。

我们使用OPNET Modeler来进行计算机仿真，将该算法和未分配优先级的二进制指数回退算法进行比较，比较这两种算法的接入延迟。表5.1列出主要参数。

表5.1 仿真参数设置

上行数据速率 (upstream_data_rate)	5.12Mbps
CM总数量	50个
微时隙尺寸 (mini-slot size)	16byte
数据包大小 (packet size)	64byte
时间/微时隙 (second per mini-slot)	25 $\mu$ s
微时隙/秒	40,000个

仿真时间 (MAX_SIMTIME)	1,200,000个微时隙, 约30秒
Ranging Backoff Start	4
Ranging Backoff End	6

我们将用户分成两个优先级: 优先级0有20个用户, 优先级1有30个用户。发出申请的用户数目都服从均值为10的泊松分布。每次用户申请的数据量为64byte。每个优先级的最低竞争时隙数为3。

这两种算法的网络负载和接入延迟关系曲线如图5.7所示。分析仿真结果, 我们可以发现, 高优先级用户的接入延迟是最小的, 这是因为基于优先级的冲突解决算法使高优先级用户尽可能地避免了初次接入延时。当网络负载比较小的时候, 三种情况的延时都比较接近, 这是因为有足够多的竞争时隙可供分配。当网络负载逐渐增加以后, 高优先级用户的延迟增加不多, 而低优先级用户的延迟开始大幅度增加, 这是因为此时竞争时隙只能保证高优先级用户的窗口大小, 而牺牲了低优先级的用户。当网络达到高负载时, 低优先级用户的延迟非常大, 而高优先级用户的延迟也有较大幅度的增加, 这是因为此时竞争时隙已经不能满足高优先级用户的窗口大小了, 而且低优先级用户只能以最低竞争时隙数来保证最小的接入。因此, 本算法充分保证了高优先级用户的接入, 在大部分情况下能够保证一个较低的接入延迟。

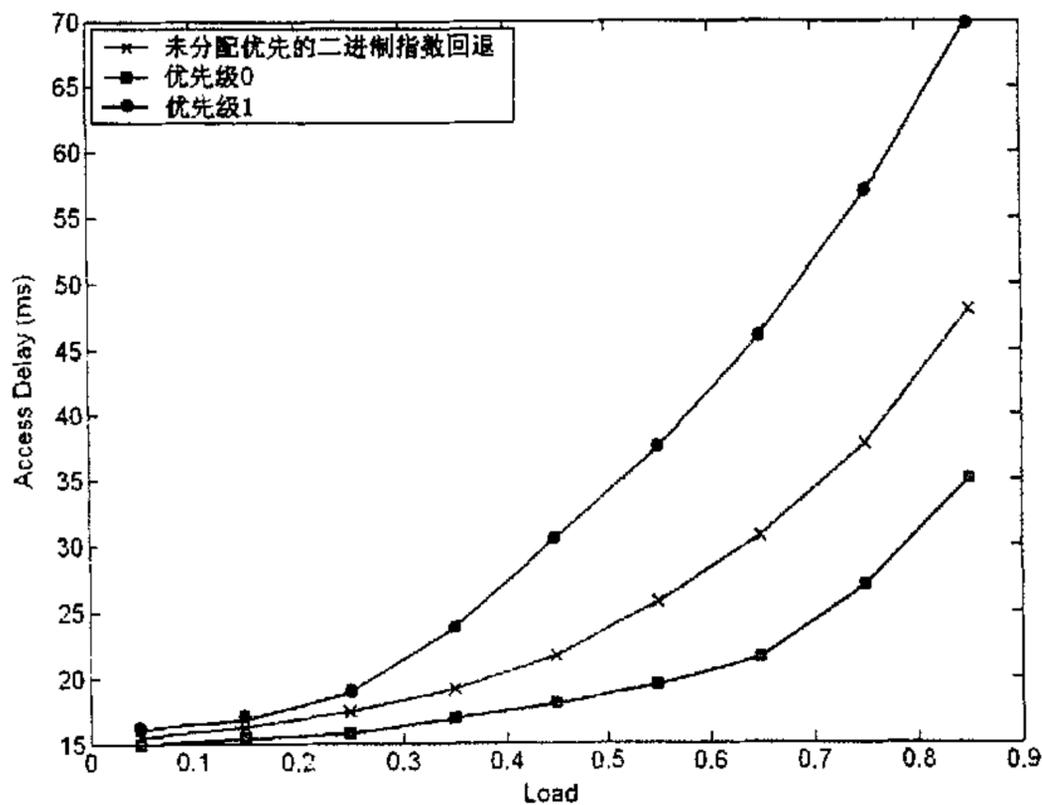


图5.7 两种算法的关系曲线

#### 5.4. 小结

本章分析了竞争时隙和延时的关系, 发现了既要按需要分配窗口大小又要减少首先接入延时这一个矛盾。为了解决这个矛盾, 设计了基于优先级的冲突解决算法。本算法使高优先级用户很好地解决了这个难题, 按照优先级的高低, 既保障了高优先级用户的接入, 又尽可能地服务于低优先级用户。

## 第6章 动态时隙分配研究

### 6.1. 研究思路

在第四章研究基于冲突检测的竞争解决算法中，我们推导出了在一定条件下（即一定的竞争时隙和发送请求的用户个数），对于一个竞争时隙而言，不同个数的用户选中这个时隙的概率。由公式（4-1），我们来观察三种情况：用户个数始终为60，竞争时隙个数分别为10，60，80，如下图所示。

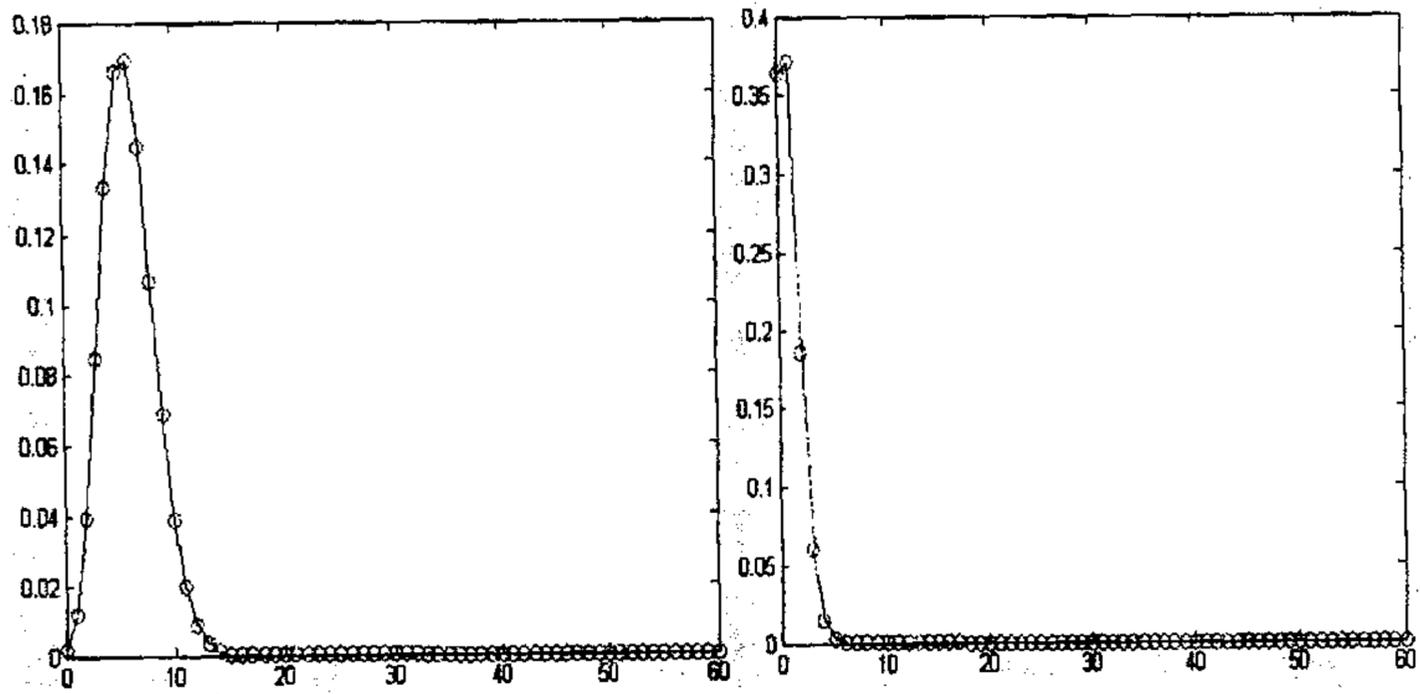


图6.1 用户个数60, 竞争时隙个数10

图6.2 用户个数60, 竞争时隙个数60

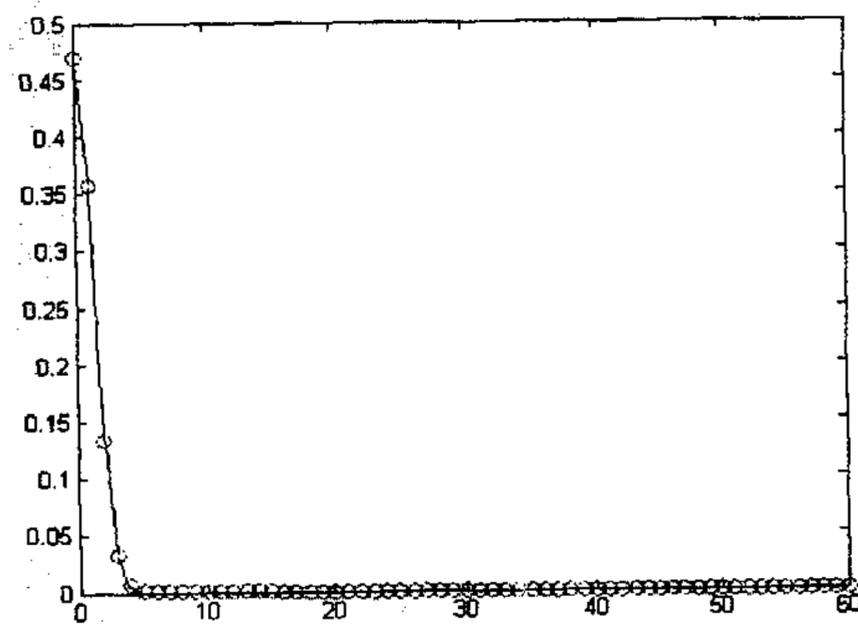


图6.3 用户个数为60, 竞争时隙个数为80

当用户个数为60, 而竞争时隙个数只有10的时候, 最大可能是有6个用户同时选中这个时隙, 而只有一个用户选中的概率只有大约1%。而当竞争时隙个数有80个的时候, 只有一个用户选中它的概率就变成了大约37%, 而有将近50%的可能性是这个时隙将被闲置不使用。虽然这三张图只是直接反映最大可能的CM个数, 但是, 换一个角度来分析, 这其实反映的是当前条件下冲突的剧烈程度。第一种情况下, 冲突非常剧烈。每一个竞争时隙最大可能是被6个CM选中而发生冲突, 只有极少的可能性能够被成功使用。而在第三种情况下, 冲突就十分微小, 竞争时隙大部分都被成功使用或者闲置。

冲突直接导致的后果就是CM无法将自己的上行带宽请求信息传递到CMTS。第一种情况下, 60个用户中, 几乎很少能有将申请传送出去的。而在第三种情况下, 60个用户中, 就可以有许多用户传送了自己的带宽请求。所以, 冲突程度直接影响了CMTS接收到的带宽请求信息的多少。而带宽请求的多少又直接导致了CMTS对上行带宽的分配使用, 换句话说, 就是直接影响了上行信道中数据时隙的数量。

## 6.2. 互动过程

通过以上的分析, 我们发现竞争时隙和数据时隙之间存在互动关系。在某些情况下, 对于CMTS而言, 一次最多能够分配的总的时隙个数往往是一个大致固定的值。<sup>[16]</sup>我们假设现在网络已经处于一个比较平稳的运作时期, 即发出请求的用户数目变化不会太剧烈。

首先, 当上行数据量比较小的时候, CMTS就会分配比较少的数据时隙, 则会剩下较多的时隙作为竞争时隙。根据前面一节的分析, 这时, 就会有较多的用户竞争成功, 向CMTS发送了带宽请求信息。

当CMTS接收到的带宽请求信息增多, 那它就必然要分配更多的数据时隙来满足增加的带宽请求。这样, 就直接导致了竞争时隙地减少。竞争时隙减少后, 由于当前用户数目变化不大, 所以竞争时隙的冲突就加剧了, 从而竞争成功的用户数目减少了, 发送的带宽请求信息也就减少了。

这个时候, 前面已经分配了数据时隙的用户已经开始使用捎带请求机制(Piggybacking Mechanism)。捎带请求机制是指: 如果CM通过“请求/授权”机制获得了上行数据时隙, 并且在传输数据之前又有新的数据到达, 则可以把

新的带宽请求附加在现在即将发送的数据帧头, 而不需要再通过“请求/授权”机制。捎带请求机制保证了CM的带宽请求能够无冲突地到达CMTS, 并且减少了系统中的冲突请求次数。所以, 减少的竞争时隙对于这些用户没有影响, 他们可以继续发出自己的带宽请求, 并且CMTS会继续为他们分配数据时隙, 而不会有其他的用户来与他们竞争。

当使用捎带请求机制的这部分用户传送完上行数据以后, 由于前面的竞争时隙冲突加剧, CMTS没有收到太多的带宽请求信息, 所以上行数据量再次减小, 可分配的竞争时隙就再次增多, 冲突就变小, 又会有较多的用户竞争成功。

因此, 竞争时隙和数据时隙其实是处于了一个动态过程中。这个过程是以上行数据量为纽带, 竞争时隙和数据时隙不断地相互影响, 相互制约, 此消彼长。这样, 就可以在不需要人为干涉的条件下, 使竞争时隙和数据时隙不断地自我调节, 以适应不断出现的新情况, 提高网络运作的效率。

### 6.3. 动态时隙分配

#### 6.3.1. 原理

在现在的网络中, 存在着大量周期性产生固定长度的实时数据的网络业务, 比如VoIP。这种数据业务称为恒定数据业务 (Constant Bit Rate, 简称CBR)。这类业务具有以下几个特点:

- (1) 以一定的周期产生数据包;
- (2) 每个周期的数据包长度相对固定;
- (3) 具有较强的实时性要求。

为了满足这种网络业务, 根据它的特点, CMTS也就必须在上行信道中周期性地分配相对固定长度的数据时隙。所以在CBR的网络业务环境中, MAP帧是周期性发送, 而且, 每个MAP帧的数据时隙个数都大致固定。因此, CMTS一次最多能够分配的总的时隙个数往往是一个大致固定的值。在这种情况下, 我们使用上一节分析的动态时隙分配, 就可以使竞争时隙和数据时隙个数自我调节, 根据实际申请的业务量动态地分配时隙, 提高网络运作的效率。

将6.2节的理论分析转化为数学的表达式, 即为:

$$\text{竞争时隙个数}(C) = \text{总的时隙个数}(M) - \text{数据时隙个数}(D) \quad (6-1)$$

需要注意一个问题, 如果只是按照式 (6-1), 则可能出现竞争时隙个数为 0 的情况。我们在第五章的分析中指出, 竞争时隙个数不能为 0, 必须要有一个“最低竞争时隙数”来保障用户最小的接入。所以, 竞争时隙个数其实是由两部分组成的: “最低竞争时隙数 ( $C_0$ )”和“分配竞争时隙数 ( $C_1$ )”。“最低竞争时隙数”是一个固定的最小值, 而每次参与动态过程的是“分配竞争时隙数”。因此, 最后该动态过程转化为数学公式:

$$C_1 = M - D - C_0, \text{ 其中 } D \leq M - C_0 \quad (6-2)$$

每次 CMTS 先根据收到的上行带宽请求分配数据时隙, 再按照式 (6-2) 分配竞争时隙。

需要注意的是, 当网络的主要业务不是 CBR 时, 使用这种大致固定总时隙长度的动态时隙分配的效果提高并不一致。首先, 当数据量很大时, 数据时隙分配较多, 与数据量较大的 CBR 的情况一致。其次, 当数据量较小时, 可能就会产生过多的竞争时隙, 其中许多都会被闲置, 所以造成延时增大。所以, 网络运作效率提高有限。

### 6.3.2. 仿真和讨论

我们将该算法和固定竞争时隙的 MAP 帧分配算法做计算机仿真, 比较这两种算法的接入延迟。比较的对象分别选用竞争时隙数目为 64 和 32, MAP 帧时隙最大为 300 个, 用户总数为 32 个, 每个用户申请的数据包大小为 64 byte。主要参数设置和 4.5.2 一致, 见表 4.3。

这两种算法的网络负载和接入延迟关系曲线如图 6.3 所示。分析仿真结果, 我们可以发现, 在低网络负载时, 三种情况下的延迟都很接近, 这是因为用户接入较少, 冲突较少。当网络负载有所增加, 用户接入数目开始增多时, 竞争时隙为 32 的延迟明显增加, 而动态分配和竞争时隙为 64 的延迟增加很少, 并且两者很接近, 这是因为当前通过动态分配还能够得到足够的竞争时隙。当网络负载继续增加, 动态分配的延迟开始明显增加, 这是因为数据时隙的大量增加导致了竞争时隙减少。因此, 通过仿真, 动态时隙分配的变化特点表现与理论分析基本一致, 在中低网络负载的条件下, 动态分配能够达到与大数目竞争时隙分配相似的延迟特性。

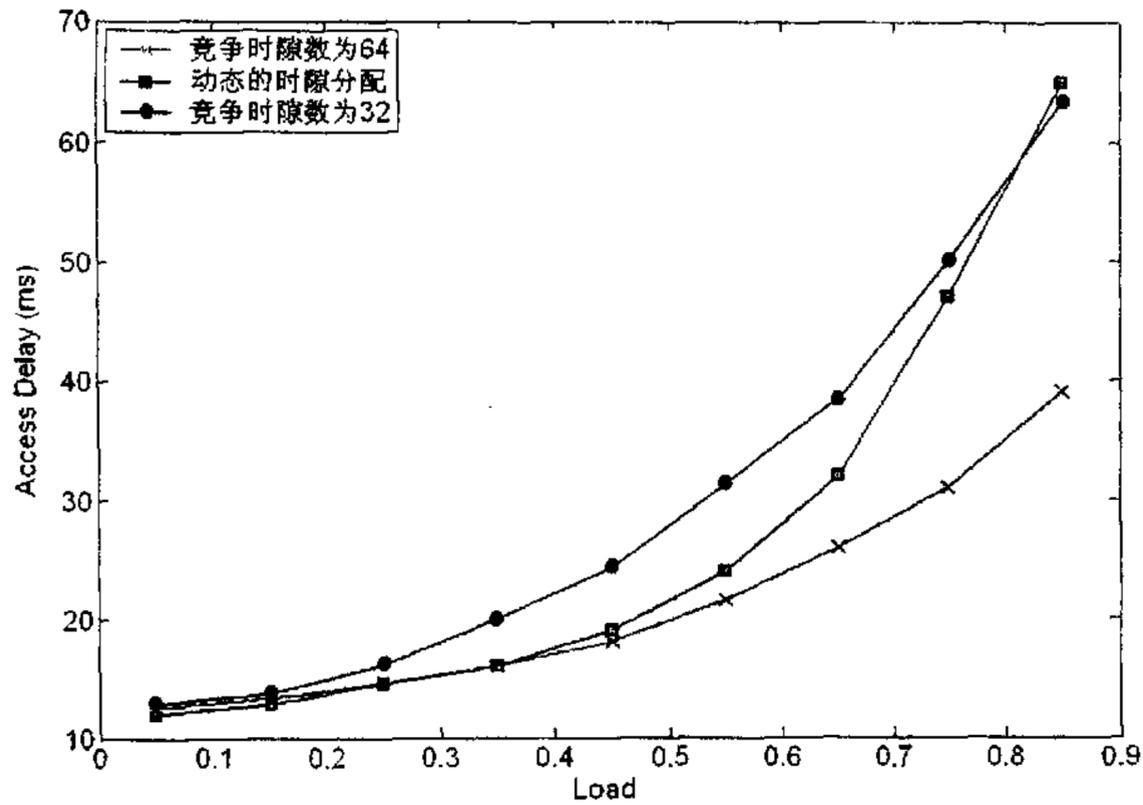


图6.3 两种算法的关系曲线

#### 6.4. 小结

本章分析了数据时隙和竞争时隙的互动关系, 提出了它们之间存在的一种动态调整关系。根据这个关系, 设计了动态的时隙分配算法, 验证了该关系在网络运作中的实际作用。但由于对它们之间这种互动的抽象总结只是一个不太准确的数学模型, 而且在大网络负载下还可能存在一定问题, 所以需要在接下来的工作中做进一步的深入研究。

## 第7章 全文总结

HFC宽带接入网的上行信道由于其特点,一直是整个网络结构中的关键和难点所在。因此,研究和设计上行信道的关键技术对整个HFC宽带接入网有着重要意义,这也是许多研究者和本文为之努力的原因。本文对基于DOCSIS的HFC网上行信道的几个关键技术做了详细的研究,取得了以下几个方面的进展:

(1) 提出了对HFC网络的改进,从而使CMTS具有了检测冲突的能力,成功解决了HFC网络存在的三个问题,双向改造困难、噪声和无法检测冲突。设计了基于冲突检测的竞争解决算法,本算法可以根据冲突的剧烈程度,调整回退窗口的大小,有效地减少冲突,以适应用户数目的不断变化。

(2) 设计了基于优先级的冲突解决算法,成功解决了既要按需要分配窗口大小又要减少首先接入延时这一个矛盾。本算法使高优先级用户获得了低延时的接入。按照优先级的高低,既保障了高优先级用户的冲突解决和首次接入,又尽可能地服务于低优先级用户。

(3) 分析了数据时隙和竞争时隙的互动关系,提出了它们之间存在的一种动态关系。根据这个关系,设计了动态的时隙分配算法,验证了该关系在网络运作中的实际作用。

这三个方面是紧密联系而非相互独立的。从整体到局部来讲,上行信道首先根据上行数据量分配了数据时隙和竞争时隙,然后又在竞争时隙中做更细致的分配。竞争时隙的分配基于冲突的程度进行调整,当无法完全按需调整时就按照优先级优先保证一部分用户。

HFC宽带接入网的上行信道研究是一个很复杂的系统性研究。本文的研究也存在一定不足,这就需要在接下来的工作中对上述的关键技术做进一步的深入研究。

## 参考文献

- [1] Andrew S. Tanenbaum, 计算机网络 (第三版), 清华大学出版社, 1998
- [2] 窦剑等, 基于 HFC 网的宽带业务接入及解决方案, 电视技术, 2000, 9
- [3] Shlomo Ovadia, 宽带有线电视接入网从技术到应用, 人民邮电出版社, 2002
- [4] B-1070 Brussels, European Cable Modem v1.0, May 1999.
- [5] CableLabs, Data Over Cable Service Interface Specification-Radio Frequency Interface Specification 2.0(SP-RFIV2.0-103-021218), Dec, 2002
- [6] IEEE, IEEE Project 802.14/a Draft 3 Revision 1, Apr, 1998
- [7] DAVIC, DAVIC 1.5 Specification DAVIC Cable Modem (Technical Specification) Revision 3.1, Nov, 1998
- [8] Ying-Dar, Lin, An Investigation Into HFC MAC Protocols: Mechanisms, Implementation, And Research Issues, IEEE communication Surveys, Aug, 2000
- [9] Golmie, N., A comparison of MAC protocols for hybrid fiber/coax networks: IEEE 802.14 vs. MCNS, ICCCN 1999, Vol 1, 1999, 266-272
- [10] 有线双向改造投资巨大不宜追求一步到位, 电子资讯时报, 2003, 7
- [11] 高菲, 2.4GHz 频段上新一代点对点无线技术, 网络通信, 2004, 4
- [12] Douglas E. Comer, Computer Networks and Internets, 清华大学出版社, 2002
- [13] Mamdouh Droubi, Nasser Idirene, Dynamic Bandwidth allocation for the HFC DOCSIS MAC protocol, ICCCN 2000, 54-60
- [14] <http://www.opnet.com>
- [15] Wei-Ming Yin, Two-Phase Minislot Scheduling Provisioning, GLOBECOM 01, 2001, 410-414
- [16] Ying-Dar, Lin, Allocation and Scheduling Algorithms for IEEE 802.14 and MCNS in Hybrid Fiber Coaxial Networks, IEEE trans on Broadcasting, Dec 1998, 427-435

## 致谢

本文是在导师刘镰斧副教授的悉心指导下完成的, 他渊博的知识、丰富的实践经验、严谨的治学态度、精益求精的工作作风、对学科发展方向的敏锐眼光和对科学的献身精神给予了我极大的启迪和引导, 在我的心中留下了难以磨灭的印象, 他是我终生学习的榜样。刘老师在我的课程学习、研究方向、研究方法、论文写作等方面都进行了精心的指导, 使本文得以顺利完成。他对我的严格要求, 使我对科学研究的精神、方法和内在规律有了较为深刻的领会, 这些收获是我今后工作的重要基础。我为有这样一位导师而感激和自豪! 其次感谢刘老师在我攻读硕士学位期间给予的关心和帮助。

感谢成都精通网络技术公司为我提供了良好的学习氛围和方便的论文工作的条件。感谢那里的每一位老师和学长, 在论文的写作过程中, 他们提出了许多有益的见解和宝贵的意见, 本文的最终完成得益于他们的巨大帮助。

感谢我的同学何剑、周再军、冯路、董均、张欣等, 他们在我的学习期间给了我很多支持和帮助, 使我愉快地度过了研究生阶段的学习生活。

最后要深深感谢我的家人, 是他们无私地支持我完成了学业。

## 个人简历、在学期间的研究成果及发表的学术论文

### ● 个人简历

出生年月: 1979 年 12 月

2002 年 9 月—2005 年 4 月: 通信与信息工程学院, 通信与信息系统  
专业攻读硕士学位

1998 年 9 月—2002 年 7 月: 通信与信息工程学院, 2002 年获得学士学位

### ● 发表论文

马俊康 《DOCSIS 网络系统的安全实现》(已被录用, 将发表在《电讯技术》2005 年第 2 期上)