DISCRETE MULTI-TONE-BASED COMMUNICATIONS IN THE REVERSE CHANNEL OF HYBRID FIBER-COAX NETWORKS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Krista Susan Jacobsen
August 1996

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

John M. Cioffi
(Principal adviser)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Robert M. Gray

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Donald C. Cox

Approved for the University Committee on Graduate Studies:

Abstract

In recent years, hybrid fiber-coax (HFC) networks have been deployed by both cable television operators and telephone companies to transport information to and from subscriber premises. Analog transmission of television signals in the downstream direction (from the operator's headend to customer premises) is common today; downstream digital transmission is expected to be relatively straightforward because downstream channels are generally high-quality. Hence, transmission using a simple single-carrier modulation in broadcast mode is probably a sufficient solution. Upstream transmission, however, presents a more challenging problem. The reverse channel of HFC networks, which has been reserved for upstream communications since the early days of cable television, is defined as the band from 5 MHz to 42 MHz. This bandwidth is often degraded by a number of severe transmission impairments, including passband ripple, spectral nulls, impulse noise, and radio-frequency ingress. Consequently, the reverse channel is seldom used today. To enable reliable, high-speed upstream transmission in the limited reverse channel bandwidth, a spectrally-efficient, robust transmission scheme is required. Furthermore, because HFC networks are generally configured in tree-and-branch topologies, the reverse channel bandwidth is shared among many users, potentially thousands, and coordination of remote unit transmissions via a channel access protocol is necessary to ensure the channel is used efficiently.

This dissertation proposes Discrete Multi-Tone (DMT) modulation for reverse channel transmission in HFC networks. The HFC reverse channel is first characterized, and an example channel is modeled for use in evaluating the proposed solution. Next, the expected performance of DMT in HFC reverse channels is quantified to illustrate that DMT achieves high spectral efficiencies and offers robustness to severe channel impairments and noise. Next, the concept of Synchronized DMT (SDMT),

first proposed by Cioffi in [1], is reviewed. SDMT is achieved when transmissions from multiple DMT-based remote terminals in a multipoint-to-point environment are synchronized so that the signal arriving at a central site receiver appears to have been transmitted by a single remote terminal, which enables the receiver to demodulate the incoming signal as it would in a point-to-point environment. SDMT dramatically reduces the receiver complexity when DMT is used in multipoint-to-point environments. Procedures for installing, synchronizing, and training remote terminals on a Synchronized DMT network are then presented. Finally, a comparison of a number of general channel access protocol families reveals that reservation-based protocols are compatible with SDMT and offer a number of advantages for HFC reverse channels. New reservation-based protocols designed specifically for multicarrier remote terminals are then presented, and their performances are quantified via analysis and simulation. It is concluded that use of SDMT-based remote terminals controlled by a reservation-based channel access protocol provides an efficient, robust solution for HFC reverse channel communications.

Acknowledgements

Because the acknowledgements will be the most widely-read part of my dissertation, I began writing this section before any other part of the thesis. Even with the head start, however, this troublesome section was the last I finished. Sometimes I was able to find just the right words to express my gratitude to those who have helped me, directly or indirectly, as I earned my Ph.D.; other times, the words I chose sounded clichéd and hollow on review. Some people have contributed so substantially to my progress as a student and as a person that a few words here cannot possibly convey what their support, encouragement, and love have meant to me. Nevertheless, I would like to take this opportunity to attempt to thank everyone who has positively impacted my Ph.D. experience.

First and foremost, to my research adviser John Cioffi—"The Man, The Myth, The Legend"—I owe my deepest gratitude. I could not have wished for a more supportive, loyal, or good-natured adviser. Since the first time I met him in 1989 when I visited Stanford as a prospective student, John has taken care of me. He offered advice about Stanford, encouraging me to attend and writing a recommendation letter supporting my application to the Electrical Engineering department. When I was accepted, he offered to be my research adviser and to supplement my fellowship stipend. At the time, not knowing how difficult it is to find an adviser at Stanford, I didn't understand the significance of the offer—I only appreciated John's confidence in my abilities and knew I wanted to work for him. Of course I gratefully accepted his offer to become one of "John's Kids." Early on, I learned that John has a great sense of humor, and I often took advantage of his good nature by "modifying" his computers at Stanford and Amati. (Changing the desktop background on his Amati PowerBook from plain blue (boring!) to a bathroom tiles pattern, which John *hated* but didn't know at first how to fix, and adding the bungee-jumping cows as his screen saver at Stanford were

two of my finest computer capers.) I tested John's good nature further by decorating his office, with the help of several officemates who shall remain nameless, for various occasions. (I think the life-sized Counselor Troi we gave John this past Valentine's Day was the crowning glory.) As my education at Stanford continued (and John refrained from ringing my neck), so did John's support and encouragement. Always ready with a pep talk when I needed one, John never seemed to doubt my ability to succeed. Events that looked catastrophic to me didn't seem to bother John at all. Through everything, he remained loyal. John, from the bottom of my heart, thank you for everything: thank you for believing in me, for taking such good care of me all these years, and for enabling me to see the world. I will never forget what you have done for me. (Hey, John, remember that million dollars I offered way back when? Well, I'm assuming we're square on that deal. With inflation and all, what's another million to you anyway?)

Although John Cioffi was by far the most influential Stanford professor as I earned my Ph.D., a number of other faculty members generously contributed their time and expertise during my graduate school years. First, Professor Bob Gray readily agreed several years ago to serve as my associate adviser, which meant he had the misfortune to sit on *both* my orals and reading committees. Dr. Gray was always accommodating, never complaining about the numerous recommendation letters I asked him to write over the years, even when the deadlines were tighter than either of us would have liked. I especially appreciated his efforts to read my dissertation promptly (within six days, to be exact) during a very busy summer quarter. Dr. Gray, thank you for all your help and support during the past several years. You are one of the most dedicated professors I've ever met. Professor Don Cox became my third dissertation reader after I caught him in a weak moment (and promised not to give him a dissertation until 1996!), and I'd like to thank him for accepting the post and consequently forcing me to clarify various explanations in this dissertation. The presentation of ideas is undoubtedly much stronger because of Dr. Cox's suggestions. Professor Greg Kovacs chaired my orals committee, and his introductory speech at my defense was one I don't think anyone who attended will soon forget. I could not have asked for a chairman better suited to my personality. Dr. Kovacs, thank you for putting me at ease on one of the most stressful days of my graduate student career. I'd also like to thank Professor Sim Narasimha for agreeing, on short notice, to be the fourth examiner on my defense

committee. In addition to acknowledging my reading and orals committee members, I would also like to thank Dr. Fouad Tobagi, who kindly allowed me to audit two of his courses and did not hesitate to offer advice on my research topic when I sought his help. His insights and suggestions proved invaluable as my work progressed. Dr. Tom Cover ("Are you going to play softball this year, or are you just going to sign up?") also helped me quickly and enthusiastically whenever I asked him probability questions. Finally, I would like to thank Dr. Bernard Widrow, who offered seemingly-endless encouragement during my first year at Stanford. Somehow, the prospect of passing the qualifying exam and earning a Ph.D. always looked less bleak after one of Dr. Widrow's pep talks. (And he has excellent taste in cars!)

In many respects, earning a Ph.D. is similar to working in a "real" job, and, as in any job, the support staff provided invaluable help and resources. In particular, our group secretary Joice DeBolt stood ready to answer questions and offer advice about work-related problems. Charlotte Coe, Mieko Parker, and Denise Cuevas also assisted our group as needed, particularly when Joice was unavailable. Joice, Charlotte, Mieko, and Denise, thank you all for your help, advice, and pleasant conversation during my stay in ISL.

I thank the National Science Foundation for the three-year fellowship they bestowed upon me. Without their financial support, in all likelihood I would not have been able to attend Stanford. Thanks also to IBM for an additional year of fellowship support and for donating "bugs–bunny," the RS6000/590 that was my most valuable tool during the last two years of my research. (Yes, you young whippersnappers reading this sometime in the future (if I may be so presumptuous to assume someone will actually read this someday), I used an RS6000. It was a good computer, so stop your snickering!) Although I greatly appreciated the speed with which bugs–bunny allowed me to run my simulations, perhaps more valuable was the CD player, which enabled me to concentrate on my work when so many more interesting things were going on in Durand 112.

Those distractions in Durand 112 were direct consequences of the office inhabitants, my dynamic, intelligent, and (thankfully) sociable officemates! They and the other research group members helped to shape my work and my world. In alphabetical order, thank you to: Naofal Al-Dhahir, Phil Bednarz, Kok-Wui Cheong, Pete "Spanky-Pie" Chow, Ivo "Knievo" Dobbelare, Minnie "MINNIEEEEE!" Ho, Joe

Lauer, Inkyu "Ink-mon" Lee, Susan "The Susanator" Lin, Acha Leke, Hui-Ling "The Lingstress" Lou, Cory Modlin, Greg Raleigh, Nick "Possum" Sands, Jose Tellado-Mourelo, Paul Voois, Rick "Heeeey, Rich!" Wesel, Katie Wilson, Zi-Ning Wu, and Nick Zogakis. I am grateful to have met and worked with everyone on this list. As my colleagues, they have taught me much about technical issues. Perhaps more importantly, however, several group-mates have become good friends whom I would like to acknowledge in a more personal manner.

Susan Lin is my closest friend from the research group. We began as "sports buddies," the only women in ISL willing to schlep to the weight room twice a week during lunchtime. (If you don't believe it, just check out Susan's HUGE muscles!) We also spent time together running, hiking, and playing volleyball, tennis, racketball, softball, and any other sport one of us could convince the other to play. At some point, however, Susan also became a great friend and confidant. I admire her boundless energy, spirit, strength, and loyalty. (And let's not forget Susan's catering skills, which were displayed elegantly by the food she arranged for my orals.) Susanator, you are one of the reasons I have enjoyed my time at Stanford so thoroughly. I value our friendship tremendously, and I know we will continue to be great friends after we graduate. I wish you the best—you deserve it.

Nick Sands was one of the first people I met in John's research group. We became friends almost immediately. I always appreciated Nick's sense of humor, his knowledge of and (admit it, Possum!) fondness for "Are You Being Served," his sensitivity to my pre-Quals fears, and his advice on everything from school to boyfriends. Possum, thank you for partaking in our Stuart Smalley "Daily Affirmations" while we were both at Stanford, and, more importantly, thanks for being such a great friend.

And to group-mate and friend Rick Wesel, thank you for your enthusiasm and unflagging willingness to help me with questions as I struggled with my research problems. I have no doubt you will be a fantastic professor and a famous researcher, and I wish you the best of luck at UCLA.

In addition to friends from within John's research group, a number of other Stanford students have become good friends during my stay. Jean Brittain ("The Jeanstress!"), who is, as far as I know, the only other woman from our year who stuck around for a Ph.D. in ISL, has been a wonderful friend. Another good friend has been Greg Plett ("It's The Greg!"), with whom I suffered through (oops, I mean,

"enriched myself in") numerous classes and studied for the dreaded qualifying exam.

During my last few years of graduate work, I was fortunate to have the opportunity to consult for Amati Communications Corporation. The experience was invaluable, and I will always treasure the knowledge I gained and the friends I made. In terms of technical expertise, the group at Amati is exceptional. Furthermore, they are a fun-loving bunch of people who maintain a friendly and exciting work environment. The personalities and intelligence of the Amati crew and my sense of loyalty made Amati my first choice for employment after graduation. Fortunately, my boss, Jim Hood, was kind enough to tell me several months ago that he wanted to hire me. After much recruiting by Jim Steenbergen (whose recruiting technique was mainly to find out where in the country my other job alternatives were and then tell me why I wouldn't want to live there) and Jim Hood (who adopted the more traditional recruiting technique of actually describing the type of work I'd be doing), Amati offered me a position as Senior Systems Engineer, which I gratefully accepted after minimal haggling. (I don't think wheeling and dealing is my calling!) As I embark on my career with Amati, I look forward to working with many of the people who have helped and influenced me during my Ph.D. In alphabetical order, many thanks to: Mike "HR!" Agah, "Jolly" Jim Aslanis, Gary Baily, Tac "Slick" Berry, John A.C. (Adonis Confucius) Bingham, Ron "Meany" Carlini, Jacky Chow, Pete "Spanky-Pie" Chow, John Cioffi (my Idol), Toni "The Tonestress" Gooch, Joe "Bon Soir!" Grady, Tan "The Man" Ha, Alan "uuh!" Hagedorn, Jim Hood, Liz "LiZZard LipZ" Kehoe, Bill Kellogg, Tim "The Timinator" Kreps ("piping hot!"), Shone Lee, Mark "Marky-Pie" Lyon, Mark Mallory, Valerie "Hearty vegetable soup" Morgan, Hong "Hongstress" Nguyen, Debu Pal, Rupa Pal, Lyle Rennick, Nick Rottler, Nick "Hallo, Possum!" Sands, Jim Steenbergen (who needs (a) a lesson about the geographic regions of the United States to learn that Colorado is NOT the Midwest, (b) to remember, each time he calls me a nerd, that people in glass houses with HP calculator collections should not throw stones, and (c) to be reminded that he *promised* me a ride in his plane, and I'm not planning to forget it!), Po "The Poster" Tong, Lalo "Where's Waldo?" Valdez, Van "The Man" Whitis, Tom Wynne, and "Big" Nick Zogakis. In addition to the individuals I've mentioned above, the Amati family includes several new employees and the Amati East folks, many of whom I have not yet met. I know, however, that many of the Amati East people, particularly those in finance

and administration, have attended to the details of my employment, and I gratefully acknowledge their efforts on my behalf. Finally, I'd like to acknowledge several ex-Amati employees, including Ang Granzotti, Ron "Ronster the Monster" Hunt, Jan Lennon, and Jim Ottinger.

Of all co-workers at Amati, I have interacted most with John Bingham; he has been my mentor as well as a supporter and friend. Acting as an unofficial second adviser, John always took time out of his busy day to let me discuss ideas with him. Consistently cheerful and interested in my work (and always ready with a good pun), John offered advice and insights that proved invaluable as my research progressed. In addition, John and I worked closely together at numerous IEEE 802.14 meetings, and I learned a great deal about both technical issues and group politics by listening to and observing John. His technical skills, inquisitive nature, sense of humor, and dedication are incredible. John, thank you for being there, as my sanity check and as a cheerleader. I hope someday to be as skilled an engineer as you are. I am honored to have worked with you and look forward to future collaborations.

Another co-worker, Liz Kehoe, has become one of my closest friends during the past two years. We always seem to have fun together, whether rollerblading, skiing ("Hood?? Good??"), horseback riding, shopping, or just sitting and talking. Often we have the same thoughts, which should cause Liz to be concerned! LiZZard, I'm glad we've grown so close, and I know we will continue to be great friends in the future. (I also know you'll continue to force me to eat my vegetables! "You're so *sallow*!") Lalo Valdez is another co-worker who has become a good friend. Liz and I have decided that Lalo ("Where's Waldo??") is an "honorary woman" (which he knows is a compliment) because he's sensitive and trustworthy. Lalo, thank you for your attentiveness to both personal and work-related issues these past two years. I am lucky to have a loyal and trustworthy friend like you. Joe Grady is one of those rare people who never seem to be in a bad mood. Joester, your sense of humor and boundless energy make you a super traveling companion and a great friend. Thanks for all your advice and constant support, especially while I was trying to figure out my job alternatives. I think the world of you and have only one question: "Would y'all like s'more NATS??" (And remember, "it used to be a svinging town, but now—not so much.") Finally, to the wine-tasters (Jim, John and Lu, Pete, Toni and Rich, and Van and Laurie), thank you for including me in your group. I thoroughly enjoy

the parties, even though my ignorance about wine means I can often only contribute comments such as, "Yuck! This one tastes like dirty socks!" or "WHOO! That one went right up my sinuses!" to the wine analysis.

In addition to those who have impacted my graduate school career while at Stanford, my family and numerous friends have supported and encouraged me since before I arrived at Stanford. First, I would like to thank Marc Goldburg for sending me John Cioffi's name when I asked him which Stanford professors might be good matches to my interests. (And I would like to apologize officially, on the record, to Marc for telling my mom, when I was 12, that there was a "hippie" at the door when Marc rang the doorbell.) Next, I am grateful to my undergraduate adviser, Jim Schroeder, for allowing me to test the research waters during several undergraduate independent study courses and during my senior project. Jim's low-key approach and words of encouragement imparted confidence to me, without which I probably would not have had the nerve to attend Stanford.

My friends Erika and John Gurley, whom I've known since well before I arrived at Stanford, have remained steadfast friends during my years as a graduate student. I always enjoy chatting with them to find out what they're doing and just to reminisce. (Erika and I have a vast and hilarious history together, which began just after we graduated high school. Example: Sitting on the sidewalk outside Dairy Queen, eating ice cream cones while staring at the grille of a car parked perhaps one foot in front of us, Erika muses, "If this car were Christine, it would run us over." In response, Krista asks, "Why? Is she a bad driver?" Hysterical laughter (Erika) and puzzled looks (Krista) follow for what seems to Krista like an eternity until Erika finally gains enough control to swallow her ice cream and explain the joke. More hysterical laughter ensues, and our heroines are late for work because Erika claims she can't drive while laughing because her "eyes get too small," and she "can't see." The boss is unsympathetic, but he is too intimidated to scold his rowdy employees.) Eweeka and John, I miss you! Someday we *must* move back to cities within a couple hours of each other.

My family, both immediate and extended, offered constant love and support during my time at Stanford. They rejoiced in my successes and empathized when I failed. Every step of the way, I felt their presence and could not have achieved my goals without them.

To my Great Aunt Beth and Great Uncle Kurt, thank you for ensuring that I finished both undergraduate and graduate school debt-free. Your generosity enabled me to focus completely on my studies. I love you and think of you often. To my grandmother Hilda, who would be proud of me even if I were pumping gas, thank you for your support, advice, and good wishes. I love you.

My brothers and sister have always played a large role my life. Although we are all very independent, I know my siblings are there for me, as I am for them. As the years pass, we seem to grow closer, even though the physical distances between the four of us continue to increase. During my first year of school, which was also my first year away from home, my brother Eric always seemed to call, just to say hello, when the loneliness was almost unbearable. His timing was uncanny. Earwig, you will never know how much those calls meant to me. I am so glad you and Jen were able to be here for graduation. My sister Britta and I have always been very close. As little girls we spent lots of time together, playing and horseback riding. Now I don't get to see my sister more than once or twice a year, but I think of her often and know that we remain great friends. For several years, we have exchanged e-mail daily, which is always a welcome break from my work. Bea, I am lucky to have you as my sister. My brother Curt has provided me with invaluable computer help over the years. Thanks to Curt, a number of Macintosh problems were solved quickly, saving me time and frustration. In addition, Curt has enriched my life with humor. During my Ph.D. I have received hilarious e-mail and snail-mail "from" all three family dogs. I've also answered the phone a couple of times when Curt has called and have been fooled into thinking someone else is on the other end! Curtieman, your sense of humor and quick wit have given me great joy during my Ph.D. To Eric, Britta, and Curt, I love you all and thank you for your love and encouragement.

My parents have always supported me in all my endeavors. Whatever the nature of the activities I chose to pursue—artistic, athletic, or scholastic—Mom and Dad never discouraged my dreams, nor did they ever appear to doubt my ability to succeed. Equally important, my parents never pressured me to do well; they simply told me that knowing I always put forth my best effort was satisfactory to them. Mom and Dad, you have worked so hard to enable me to become the person I am today, and I appreciate the financial and emotional sacrifices you made to "raise me right." Thank you for teaching me that there is satisfaction in achieving a difficult goal, and that

quitting is far worse than failing. Finally, thank you for instilling in me the belief that I can do and be anything. That gift was the greatest of all. I love you.

Last, but perhaps most importantly, one person has influenced my recent educational and career choices more than anyone else. My dearest friend John inspired me to choose electrical engineering as my undergraduate major, saying he thought I would be a great engineer. As my studies progressed and the challenges grew, he listened patiently to my complaints, offering constant encouragement; and when I did well, he enjoyed my success as much as I did. During the difficult first year at Stanford, John's unwavering love and encouragement helped dull the pain of the all-too-frequent academic defeats. He always cheered me on, supporting my efforts, telling me he knew I could achieve my goals. John, thank you. You have influenced and shaped me so dramatically during the past eleven years that words can't possibly express how grateful I am or how I feel about you. You are an exceptional person—kind, generous, supportive, and loyal—and I am fortunate to have you in my life. We have built the sort of relationship that everyone longs for but most are never lucky enough to experience. I know we will remain best friends. I love you with all my heart, now and always.

For all the ways they have loved, supported, influenced, and encouraged me, I dedicate this dissertation with much love to my parents and to my best friend John. Without them all, I would not be the person I am today.

And to everyone who has helped to make the Ph.D. experience a delight, thank you. I will always reflect on my time at Stanford with great nostalgia.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In recent years, growing interest in transmitting information to and from homes and businesses has sparked numerous efforts to devise schemes that provide customers with high-speed, low-cost data paths. Table 1.1 gives some examples of the information services that have captured the imaginations of consumers and consequently helped to drive these efforts. The table indicates that as the sophistication of information signals grows, higher bit rate channels are required to support these services satisfactorily. Practical voice-band modems are limited to transmission rates on the order of 30 kbps, which is not sufficient to support many services of interest. *Integrated Services Digital Network (ISDN), High-Speed Digital Subscriber Lines (HDSL), Asymmetric Digital Subscriber Lines (ADSL)*, and most recently *Very high-speed Digital Subscriber Lines (VDSL)* are services that use twisted-pair frequency bands beyond the voice band to support higher data rates. ISDN (which can support up to 144

Table 1.1: Example data services

Service	Data rate		Data type
	To subscriber	From subscriber	
Video conferencing	128 - 384 kbps	128 - 384 kbps	Constant bit rate
Video-on-demand	1.5 - 6.0 Mbps	$<$ 100 bps	Constant bit rate
Internet access	$\leq$ 10 Mbps	$\leq$ 10 Mbps	Packetized
Home security	~ 0	$<$ 100 bps	Constant bit rate
Video games	$<$ 1.0 Mbps	$<$ 100 bps	Packetized
Digitized voice	64 kbps	64 kbps	Constant bit rate

kbps), HDSL (which supports up to 2.048 Mbps), and ADSL (which supports up to 8 Mbps from the telephone company central office to the home, and up to 640 kbps in the opposite direction) have been tested and deployed by telephone companies around the world, and VDSL (which delivers up to 52 Mbps to the subscriber premise) is envisioned as the next generation service for the twisted-pair environment. Hence, significant effort has been expended to increase the data rates supported by ordinary telephone lines.

In the United States and in many other countries, however, cable television plants offer an alternative path to subscriber premises. The transmission characteristics of coaxial cable itself are superior to twisted pair: the line attenuation is less severe with increasing frequency, enabling cable networks to span large distances, and the shielding of coaxial cable renders it less susceptible than twisted pair lines to many potential interfering signals. Cable television networks, however, were installed to support television signals broadcast from cable operators over large distances to subscribers, and consequently they are characterized by some unique problems. In particular, the amplifiers used to counter signal attenuation caused by coaxial cable inject noise into the network, and they require maintenance to ensure that the signal quality at subscriber premises is adequate. Amplifiers nearer to the broadcast signal point of origin must be well-maintained because any degradations they introduce will be seen by *all* subscribers on the network. To reduce network maintenance requirements, a new architecture called *Hybrid Fiber-Coax (HFC)* has evolved. In HFC networks, the coaxial cable that typically spans the largest distance is replaced by fiber, which eliminates the most problematic network amplifiers and their associated service requirements. Thus, an HFC network is typically less noisy than its all-coaxial counterpart, but HFC networks are still configured to support signals broadcast by the cable operator to subscribers; the architecture is not immediately conducive to transmission from subscribers to the cable operator. This reverse path transmission problem is addressed by this work.

1.1 Problem Statement

This dissertation considers the general problem of transmitting information on a shared channel from a set of independent *remote terminals* (or *remote units*) to a

central-site receiver when the remote units transmit using *Discrete Multi-Tone (DMT)* modulation. Hence, the problem can be considered *multipoint-to-point* transmission with DMT-based remote terminals. The particular environment addressed in this work is the hybrid fiber-coax network *reverse channel*, a portion of the cable television spectrum that is reserved for transmission of signals from subscriber premises to a cable operator's *headend*, which is the origination point of signals broadcast from the cable operator to subscribers. The reverse channel offers a limited bandwidth that must be shared by hundreds or, on some larger networks, thousands of remote terminals. The direction signals travel in the reverse channel (from subscribers to the headend) is referred to as *upstream*, whereas signals broadcast from the headend propagate *downstream*.

Cable Television Laboratories (CableLabs), in its request for proposals for a high-speed cable data service, has identified a number of design goals and requirements that an HFC reverse channel communications system should meet. [3] Among them are:

1. **Compatibility with existing HFC network designs, parameters, and services**: Most existing HFC networks continue to support the broadcast of analog video signals, and some networks may support other systems. For example, a cable operator may have installed on the network a two-way communications link that uses part of the reverse channel bandwidth. Any reverse channel transmission system should be compatible with these services. Furthermore, HFC parameters such as spectral allocations should be observed by the design.

2. **Efficient use of the limited reverse channel bandwidth**: Because hundreds or thousands of subscribers may need to transmit using the reverse channel, which is a fixed bandwidth, it is desirable that the system design maximize the number of simultaneous users.

3. **Ability to support up to 2,000 users per fiber node**: Although CableLabs estimates that most HFC networks serve approximately 500 homes, in some geographic locations a single network may be required to support as many as 2,000 remote terminals.

4. **Resilience to a variety of channel noises and impairments**: The design should ensure that service impairments do not cause any long-term degradations. Recovery from transient impairments should be automatic and immediate.

5. **Support of a unique network address at the subscriber premise**: Although each remote terminal will be assigned a unique serial number during the manufacturing process, using this typically-long serial number to communicate with a particular terminal is inefficient. Instead, a unique, shorter *node address* can be assigned to a terminal to enable more efficient headend-to-remote terminal (and vice-versa) communications.

6. **Ability to support a variety of services**: As Table 1.1 indicates, some information sources generate data at a *constant bit rate*, and others generate data in discrete *packets*. For example, a digitized voice signal must be supported, in general, by a constant bit rate channel, whereas a video game generates discrete packets at irregular intervals. Moreover, the instantaneous bit rate required by the video game source could be anywhere from zero to some peak rate, depending on the user's activity. Hence, in contrast to constant bit rate sources, the required bit rate of a packetized source may vary with time. The reverse channel HFC design should be able to support a mixture of both constant bit rate and packetized sources along with a wide range of instantaneous bit rates. CableLabs cites a minimum peak burst rate of 500 kbps, and Table 1.1 suggests that a maximum burst rate of 10 Mbps is desirable.

7. **Flexible, fluid, and, to the subscriber, transparent allocation of the available spectrum**.

8. **Ability to track subscriber usage of the return channel**: To facilitate billing, the design should enable the network operator to monitor subscriber usage.

An additional goal, as in any design, is to minimize cost and complexity while meeting the other requirements.

In this dissertation, an HFC reverse channel communications system that meets

these criteria is designed. The design uses Synchronized Discrete Multi-Tone modulation in conjunction with a channel access protocol to enable highly efficient, cost-effective digital communications in the reverse channel.

1.2 Outline of Dissertation

Before a system can be designed to provide reverse channel hybrid fiber-coax communications, it is necessary to gain an understanding of the HFC environment. Chapter 2 begins with a brief history of cable television networks and then describes the so-called "tree-and-branch" topology in which cable networks are traditionally configured. Next, the HFC network reference architecture is presented. A variety of channel impairments the HFC architecture can introduce into the reverse channel are then described. In particular, causes of passband ripple, spectral nulls, and various noises, including radio-frequency ingress, impulse noise, and additive white Gaussian noise, are identified. Finally, a model HFC network configuration is constructed, and the reverse channel transfer characteristics resulting from an analysis of the model system are extracted to evaluate Discrete Multi-Tone modulation for reverse channel communications.

Chapter 3 describes Discrete Multi-Tone (DMT) modulation, a multicarrier transmission technique that partitions a channel into a set of orthogonal, equal-bandwidth subchannels and then assigns to each subchannel a number of bits based on the subchannel's signal-to-noise ratio and the desired symbol error probability. By tailoring the bit distribution to the attenuation and noise characteristics of a particular channel, DMT avoids regions of the spectrum that are too noisy or too severely attenuated to support meaningful communications. Consequently, DMT can maximize the spectral efficiency of a transmission system. Simulation results are presented in Chapter 3 to illustrate this ability and to quantify the performance achievable by DMT on the channel simulated in Chapter 2. Additionally, a reference system is constructed for use as an example throughout the remainder of this dissertation.

In Chapter 4, *Synchronized DMT (SDMT)*, a method of synchronizing multiple DMT-based remote terminals to enable multipoint-to-point communications, is reviewed. Essentially, SDMT synchronizes transmissions from the remote terminals so that the signal arriving at a central-site receiver appears to have been transmitted

by a single remote terminal. A channel access protocol is then used to control which remote unit transmits at any given time or in any given frequency band. The chapter begins with a discussion of DMT modulation in a multipoint-to-point environment. The need for synchronization is illustrated by a simple example system that uses a very restrictive and, hence, impractical channel access protocol. The example illustrates that synchronization preserves the elegant subchannel orthogonality achieved straightforwardly in the point-to-point case. Based on the synchronization discussion, it is concluded that, to minimize the design cost, remote terminals should be synchronized to transmit data on the reverse channel. A procedure for installing the remote terminals is then described. The installation process includes synchronizing a remote terminal and determining the best bit distribution for it based on the reverse channel characteristics. Finally, because the channel and noise can change as a system operates, a procedure for retraining the remote terminals is described.

Chapter 5 addresses the issue of reverse channel access control. Generalized protocol alternatives are reviewed first, and it is concluded that a reservation-based protocol is a good choice to coordinate transmissions on HFC reverse channels. Several new reservation-based access protocols designed specifically for networks that use generalized multicarrier-based remote terminals are subsequently presented and analyzed. Simulation results are compared to determine which of the new reservation-based protocols offers the best performance in terms of channel access delay (the time from when a remote terminal has a message ready to transmit and when the remote terminal begins to transmit that message) and message transport delay (the time from when a remote terminal has a message ready to when that message is received successfully at the central site). The chapter shows that protocols that take advantage of the accessibility of both the time and frequency dimensions offered by multicarrier modulation outperform protocols that operate in either a strict time-division or frequency-division manner.

Chapter 6 summarizes the results of the dissertation, discusses how well the Discrete Multi-Tone design meets the requirements outlined in this introduction, and suggests additional research topics in the area of reverse channel transmission on HFC networks.

1.3 Contributions

The author's major contribution is the system design presented in this dissertation. To the author's knowledge, many of the ideas and their application to HFC reverse channel communications are original. First, the evaluation of the performance of Discrete Multi-Tone modulation in HFC reverse channels, which is presented in Chapter 3 of this dissertation, is an innovation. Prior to this work, no one had proposed DMT for upstream communications on HFC networks or quantified its achievable performance. Because DMT is a multicarrier technique, its use in the multipoint-to-point reverse channel environment, which itself imposed some interesting constraints, created some unique problems that required creative new solutions. For example, although Cioffi's patent application for Synchronized DMT describes the conditions DMT-based remote terminals must meet to enable the use of DMT in multipoint-to-point environments, it does not specify how these conditions should be achieved. [1] Consequently, methods for meeting the requirements of Synchronized DMT on HFC reverse channels were devised by the author. The installation, synchronization, training, and retraining procedures presented in Chapter 4 were developed in this work to enable the use of Synchronized DMT. Proposing a reservation-based protocol for HFC reverse channels is not an original idea, but the multicarrier protocols developed by the author are new. In particular, the idea of partitioning a multicarrier symbol into a set of concurrent, disjoint reservation slots is, to the author's knowledge, an innovation. The analysis of the RBM-T protocol and simulations of the RBM-T and RBM-TF protocols, all presented in Chapter 5, are also contributions to the field.

A final contribution is the author's work in the IEEE 802.14 Working Group, which is standardizing communications on HFC networks. As a result of the early work on the achievable performance of DMT in HFC reverse channels, the author was invited by the chairman of what was to become the IEEE 802.14 Working Group to propose DMT as a solution for reverse channel transmission. Since that time, working with Amati's John Bingham, the author has written and presented multiple contributions for the 802.14 committee in an attempt to convince them to choose DMT as the standard.

Finally, the author would be remiss not to acknowledge the valuable contributions of John A.C. Bingham to this work. Although the research presented in this

dissertation is original, over the years the author has participated in numerous fruitful discussions with John Bingham, and those conversations and John's advice most certainly helped to shape the author's thinking and work.

Chapter 2

Hybrid Fiber-Coax (HFC) Networks

This chapter reviews hybrid fiber-coax (HFC) networks and presents a reverse channel model that will be used subsequently to evaluate the proposed design. The chapter begins, in Section 2.1, with a review of the all-coaxial cable television networks from which HFC networks have evolved. The HFC architecture, including hardware and topology, is described in Section 2.2. Section 2.3 discusses the HFC reverse channel environment. Various transmission impairments, including those introduced by the channel itself and common reverse channel noises, are described. In Section 2.4, the effects of the impairments presented in Section 2.3 are translated into a reverse channel model, which is used as an example throughout the remainder of the dissertation.

2.1 Cable Television Systems

The hybrid fiber-coax architecture is an evolution of the familiar all-coaxial cable television network. Therefore, before describing the HFC network topology, it is helpful to present some background information about cable television systems. An historical overview describes the evolution of cable networks to the current *tree-and-branch* architecture. In addition, the characteristics of coaxial cable, the core of cable television systems, are reviewed. For additional details on cable television networks, the reader is referred to [4], [5], or [6].

2.1.1 Historical Overview

Cable television networks were deployed originally for the sole purpose of delivering analog television signals to homes that could not receive over-the-air broadcast signals with an antenna. In some cases, homes were out of range of over-the-air signals, and in other cases the terrain made high-quality reception difficult. [4] Hence, the offerings of the first cable television networks were merely duplications of publicly-available over-the-air signals, but the signals were transmitted over more reliable channels, such as twin-lead lines or coaxial cables strung from home to home. Eventually, cable operators deployed structured networks composed of coaxial cables, amplifiers, and taps. In the mid-1970s, satellite delivery of signals enabled cable television operators to expand their offerings to include programming unavailable from terrestrial broadcasters. Because the cable system frequency spectrum is theoretically independent of the over-the-air spectrum, cable operators were able to add new channels by "re-using" frequencies that in the over-the-air spectrum are assigned to services other than television broadcast, as illustrated in Figure 2.1. Specialized channels for news, sports, shopping, and movies were transmitted by most cable operators in the "new" frequency bands, and subscribers wishing to receive the additional channels were charged fees each month. Essentially, today's cable systems are operated in the same manner.

In addition to detailing the spectral allocations of channels that carry signals from the cable operator to subscribers, Figure 2.1 also illustrates the *reverse channel* bandwidth, from 5 to 42 MHz, allocated for upstream communications. Even in the early days of cable television, a need for the support of transmissions from subscribers to cable operators was recognized. Unfortunately, however, as later sections of this chapter will illustrate, the bandwidth reserved for upstream communications is arguably the most hostile toward communications in the entire spectrum. For this reason, the reverse channel is still today seldom used, even though some specialized services offered on cable systems require communications from subscribers to the cable operator. An example is pay-per-view programming, where the broadcast of an event can be received (theoretically) only by those subscribers who have agreed to pay a fee to view it. Ironically, although such services as pay-per-view are offered regularly by nearly all cable operators, ordering must generally take place over an automated telephone link because the reverse channel is unreliable and rarely enabled. [4]

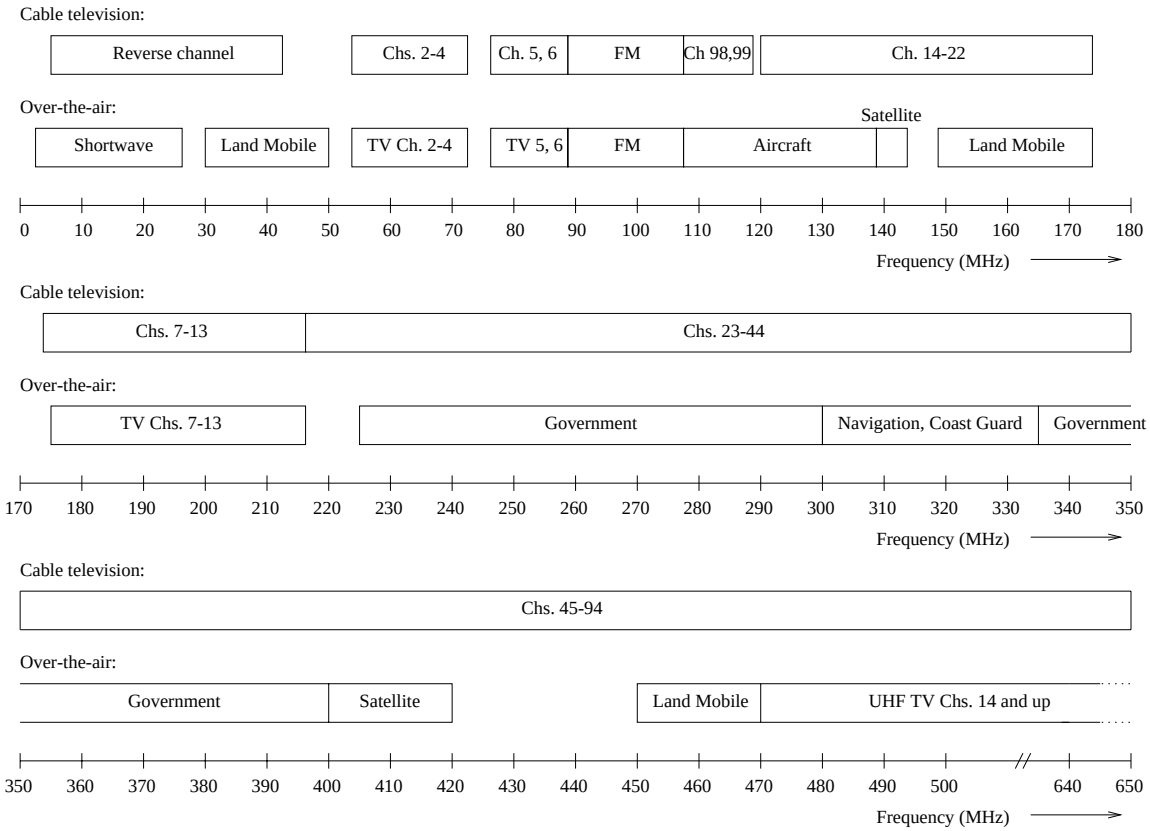

Figure 2.1: Cable television and over-the-air frequency allocations [2]

2.1.2 Coaxial Cable

The core of cable television plants is the coaxial cable that transports signals to subscribers. Coaxial cable is composed of an inner conductor (typically copper-clad aluminum or copper-clad steel), a concentric outer conductor (usually made of aluminum or a combination of metal foil and braided wire), and a dielectric that separates the two conductors, as shown in Figure 2.2. [4] The outer conductor acts as a *shield*, confining the transmitted electric field to within the coaxial cable and preventing external signals from penetrating the cable. [7] Assuming perfect conductors and a lossless dielectric, the theoretical characteristic impedance Z_o of coaxial cable is given by

$$Z_o = \frac{60}{\sqrt{\varepsilon_r}} \ln\left(\frac{b}{a}\right) \text{ Ohms},$$

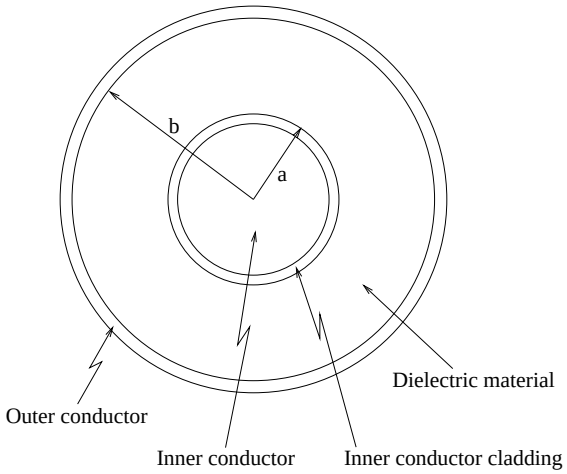

Figure 2.2: Coaxial cable cross section

where ε_r is the relative permittivity of the dielectric material, b is the inner diameter of the outer conductor, and a is the outer diameter of the inner conductor. (See Figure 2.2.) For coaxial cables used in cable television systems, a, b, and ε_r are selected so that $Z_o = 75\Omega$.

Like any physical medium that transmits electrical signals, real-world coaxial cable is characterized by a loss, which is a consequence of manufacturing cables from real-world conductors and dielectrics. The loss is a function of the cable diameter, length, and dielectric material, as well as operating frequency and temperature. Of these parameters, diameter, length, and frequency most strongly determine the attenuation of signals transmitted on real-world coaxial cables, assuming a reasonable dielectric is used. Exact equations for the attenuation are given in Appendix A, but in general at any frequency the loss is inversely proportional to the cable diameter. Hence, cables with smaller diameters incur more significant losses at any frequency than do larger-diameter cables. Furthermore, at any frequency the attenuation is directly proportional to the length of cable. Additionally, the loss is proportional to the square root of frequency. Figure 2.3 plots attenuation as a function of frequency for three lengths of 0.5-inch diameter coaxial cable (assuming a copper-clad aluminum center conductor, an aluminum outer conductor, and a dielectric with $\varepsilon_r = 1.16$), illustrating the frequency- and length-dependence of cable attenuation.

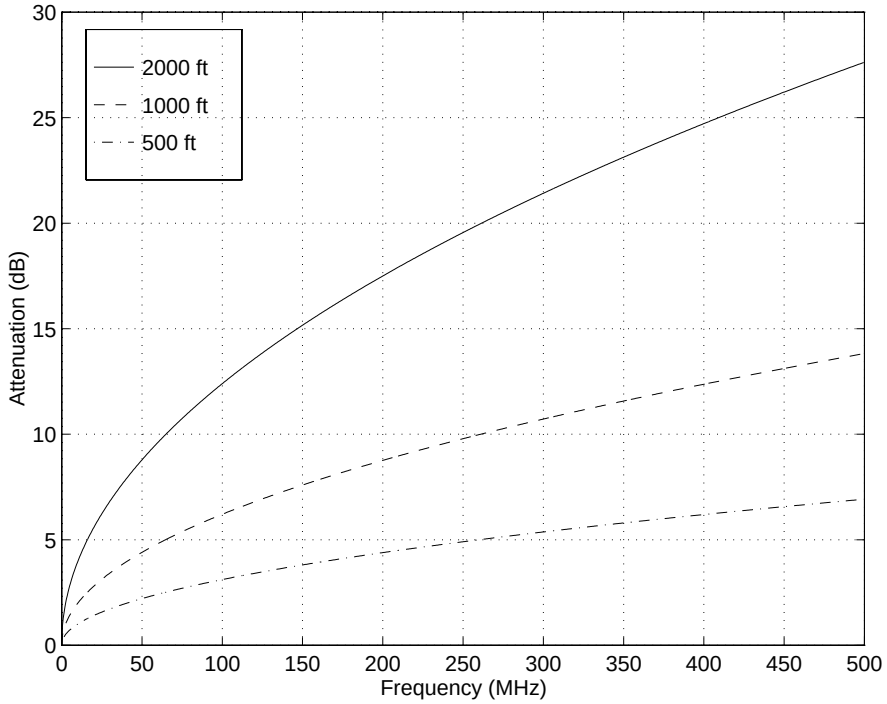

Figure 2.3: Attenuation of 0.5-inch coaxial cable

2.1.3 Cable Television Network Architecture

Figure 2.4 illustrates the topology of most cable television systems. Signals broadcast on a cable network originate at the *headend*, where both satellite and local and distant over-the-air television signals are received via antennas. In addition, some programming transmitted on the cable network may originate at the headend; examples include commercial advertisements, taped programs, and live productions from studios located at the headend. The signals are launched on *trunk cables*, which transport signals to residential neighborhoods in the bandwidth between 50 MHz and some upper frequency that may be as low as 220 MHz in small systems or as high as 750 MHz in large systems, in accordance with the channelization shown in Figure 2.1. Inserted approximately every 2,000 feet along the trunk cables are broadband *trunk amplifiers* that counter signal attenuation caused by the trunk cable. In residential neighborhoods, *bridger amplifiers* interface the trunk cable with *distribution cables*. The bridger amplifiers increase signal levels for delivery to multiple homes. Along

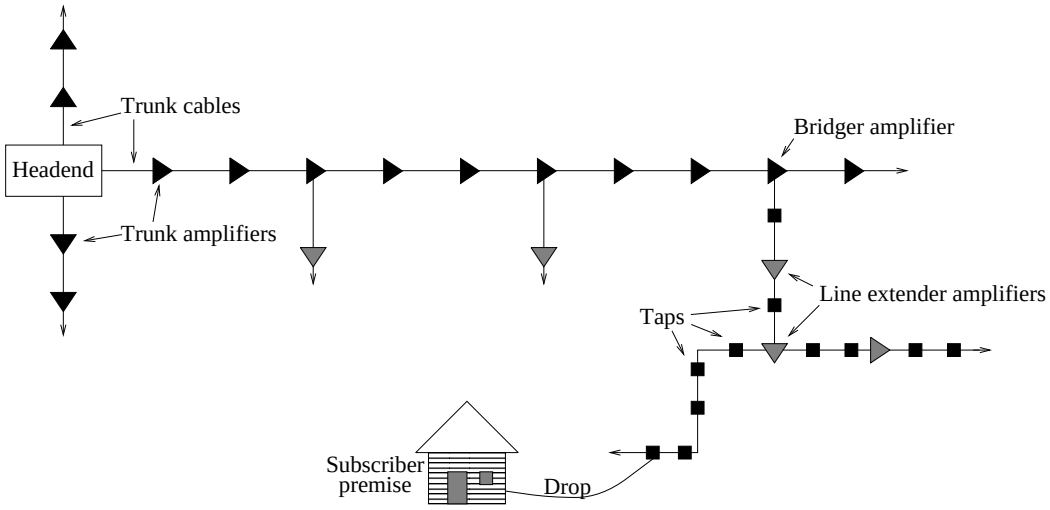

Figure 2.4: Cable television system topology

each distribution cable, one or two *line extender amplifiers* may be inserted 300 to 900 feet apart to preserve signal quality. Finally, *taps* deliver signals from the distribution cable to the familiar flexible *drop cables*, which are routed to residences and connected either to a set-top box or, if a so-called "cable-ready" television is used, to the television itself. The cable television topology is often referred to as a *tree-and-branch* configuration because the various parts of the network are analogous to parts of a tree.

Like all systems built from passive and active components, cable networks are degraded by thermal noise due to the heat-induced random movement of free electrons. [8] The minimum thermal noise power in a 4-MHz band of a perfect cable network at room-temperature is 1.62×10^{-14} Watts. Signal levels in cable systems are commonly represented in decibel-millivolts (dBmV), which is a measure of decibels relative to 1 mV across the 75 Ω characteristic impedance of coaxial cable. Given a noise level η in Watts, in dBmV the noise is

$$\text{Noise in dBmV} = 10 \log \frac{\eta}{1.33 \times 10^{-8} \text{ Watts}} \tag{2.1}$$

Hence, using Equation 2.1, the minimum noise level in a 4-MHz band of a cable network is -59.2 dBmV. This level is often called the *thermal noise threshold*.

On practical cable networks, amplifiers contribute noise in excess of the thermal noise threshold. Every amplifier is characterized by a *noise figure*, which is the amount of noise the amplifier contributes above the noise threshold. [8] Furthermore, because relatively high signal levels are required at the input to the distribution plant to ensure adequate signal quality at subscriber premises, some amplifiers on the network may be driven into regions of their transfer characteristics that are slightly nonlinear. Consequently, these amplifiers cause signal distortion that is a function of both the signal level and bandwidth. Higher signal levels and bandwidths cause more severe distortion products. [4] Both the aggregate thermal noise and distortion increase as additional amplifiers are inserted in the network. For this reason, and to reduce system costs, cable operators generally try to minimize the number of amplifiers on their networks. For example, the maximum number of trunk amplifiers on cable networks supporting large numbers of channels is usually less than thirty, whereas networks supporting fewer channels, which span a narrower bandwidth, may be able to tolerate distortion contributed by up to fifty or sixty amplifiers. [4]

Of the total footage of coaxial cable in most cable television plants, approximately 45% comprises subscriber drops, 40% is found in the distribution cables, and the remaining 15% of the footage is in trunk cables. [4] Because trunk cables carry signals over large distances, 1.0-inch or 0.75-inch diameter coaxial cable is typically used to reduce the signal attenuation per unit length and, consequently, the number of trunk amplifiers required to maintain an acceptable signal quality. As a result, the ambient noise contributed by network hardware is kept as low as possible. To reduce system cost, the diameters of distribution and drop cables are generally 0.5 inches and 0.25 inches, respectively. Smaller diameter cables can be used without requiring many more amplifiers because any particular distribution or drop cable spans a much shorter distance than do trunk cables, even though the total footage of distribution and drop cables far exceeds the total footage of trunk cables.

2.2 HFC Network Topology

Hybrid fiber-coax (HFC) networks were developed to better use the cable television infrastructure and to reduce maintenance and repair costs. Like cable television networks, HFC networks are configured in tree-and-branch architectures. However,

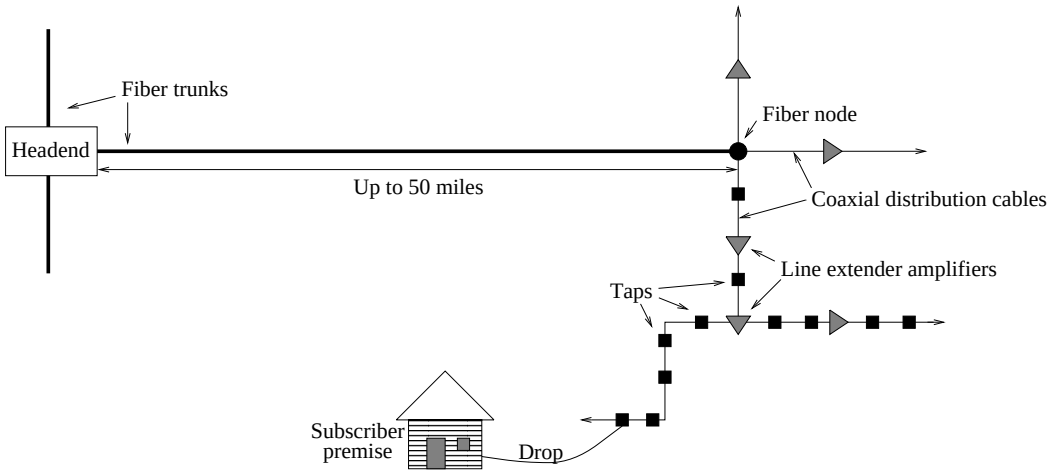

Figure 2.5: HFC network configuration

a *fiber trunk*, usually less than 50 miles long, replaces the coaxial trunk to transport signals between the headend and a *fiber node*. By substituting fiber for the trunk cable and its associated amplifiers, both the channel thermal noise and overall amplifier distortion and maintenance requirements are reduced significantly. Hence, the HFC environment provides a less noisy, lower-maintenance channel to the cable operator. At the fiber node, optical signals are converted to electrical signals (or vice-versa for signals traveling from subscribers to the headend). As in traditional all-coaxial cable systems, distribution and drop cables then transport signals between the fiber node and as many as a few thousand remote terminals. [3] Typically the maximum length of coaxial cable between the fiber node and remote terminals is 2 miles. [9] Figure 2.5 illustrates the HFC network topology.

As in cable television networks, the frequency spectrum of HFC networks is configured to support duplex transmission in disjoint frequency bands. In most existing HFC networks, known as *sub-split* systems, downstream transmission is supported in the frequency band from 50 to 550 MHz (or from 50 to 750 MHz in upgraded systems). Presently, the downstream bandwidth of HFC networks supports, in general, only the broadcast of the familiar analog cable television signals, although transmission of digital signals in the downstream bandwidth is currently under investigation by various cable operators and standards organizations. Upstream transmission in sub-split systems is confined to the frequency band from 5 to 35 MHz (or from 5 to

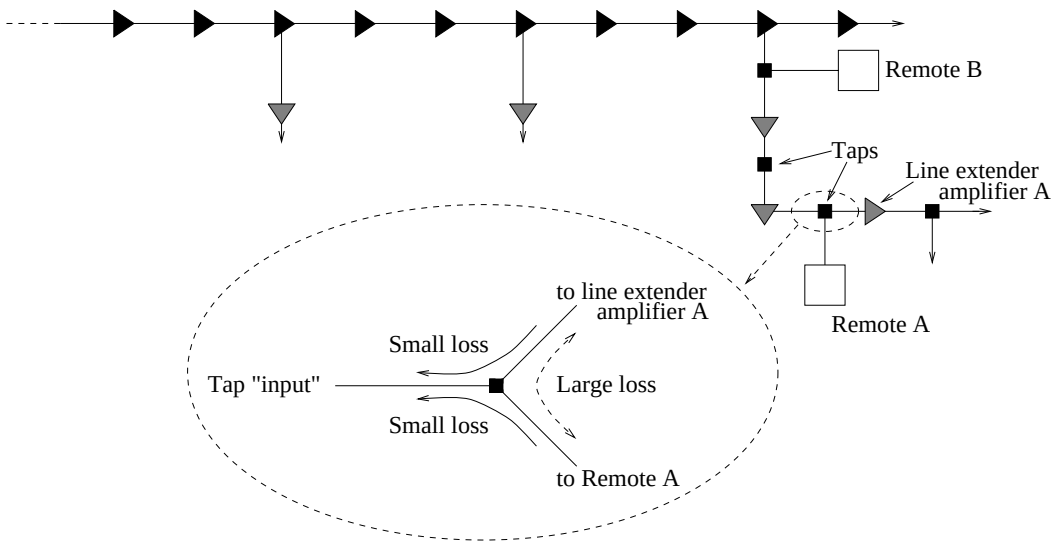

Figure 2.6: Tap characteristics render direct remote-to-remote communications impossible

42 MHz in upgraded systems). The upstream and downstream spectral allocations in sub-split networks are essentially the same as in traditional cable television networks. Newer HFC network deployments may support a *mid-split* (5 to 108 MHz for upstream transmission and above 162 MHz for downstream transmission) or a *high-split* (from 5 to 174 MHz for upstream transmission and above 243 MHz for downstream transmission) frequency band allocation. [3] Regardless of the particular bands used for upstream and downstream transmission, diplex filters and amplifiers are used to separate the two bands, and remote terminals are unable to receive transmissions in the reverse channel. Even if remote terminals were equipped with reverse channel receivers, tap characteristics would prevent them from receiving signals transmitted by other remote terminals in the reverse channel. Figure 2.6 illustrates why. Considering the tap orientation from the downstream broadcast perspective, signals transmitted "backward" through taps are attenuated only slightly between either tap "output" and the "input." In contrast, the attenuation between the two tap "outputs" is quite large, and even remote terminals connected to the same tap are unable, under normal circumstances, to receive transmissions from each other via the reverse channel. Clearly, if multiple taps separate two remote terminals wishing to communicate

with each other, then reverse channel signals transmitted by either remote are attenuated even more severely as they pass through the additional taps. For example, in Figure 2.6, neither Remote A nor Remote B can detect signals transmitted by each other in the reverse channel. This hardware limitation imposes a constraint on HFC network transmission: all remote-to-remote transmissions must pass through the headend, where they are remodulated into the downstream bandwidth. As Chapter 5 will illustrate, this requirement limits the number of reverse channel access protocol alternatives.

2.3 HFC Reverse Channel Characterization

Before a reverse channel communications system can be designed, it is necessary to understand the transmission environment and impairments. In Section 2.1.2, the frequency-dependent attenuation of signals caused by coaxial cable was described. Transmission of digital signals on HFC networks is further complicated by two additional general effects: micro-reflections and noise. Micro-reflections are delayed versions of the desired signal that are created when the original signal is reflected at points in the network at which the impedance is discontinuous. Because hardware used in HFC networks is not ideal and network installations cannot be perfect, various hardware including taps, amplifiers, connectors, splices, and splitters can all cause signals to be reflected at their insertion points. Unterminated (that is, open-circuited) coaxial stubs introduce standing waves in the channel. In addition, damage to coaxial cable (such as kinks and dents) can also cause signal reflections. The effect of these reflected signals on the frequency transfer characteristic of an HFC network is passband ripple, which, in downstream analog television channels, causes the familiar "ghosting" effect. The impact of ripple on digital signals, however, whether transmitted upstream or downstream, can be more severe. Variations in a channel's frequency response can cause successively transmitted symbols to interfere with one another, a well-known effect called intersymbol interference (ISI). Depending on the absolute deviations of the channel magnitude response (which are determined by the magnitudes of impedance mismatches) and the frequency with which those deviations occur in the frequency response (which depends on the length of transmission line between mismatches), a given symbol could interfere significantly with a large

number of adjacent symbols. Recent studies of existing HFC networks have shown that signal reflections attenuated less than 20 dB with respect to the original signal generally occur at delays of less than 0.5 μs, although more attenuated echoes can occur up to approximately 1.5 μs after the initial transmission. [10] Any digital transmission scheme used on an HFC network must be somewhat robust to microreflections, regardless of its spectral efficiency. However, efficient modulation techniques that transmit large numbers of bits per symbol must be especially resilient to ISI to protect the system against detection errors.

A second source of signal degradation in HFC reverse channels is noise. Significant effort to characterize the HFC upstream noise environment has been expended in recent years by Cable Television Laboratories and several of its member companies. As a consequence of their studies, two major noise sources, ingress and impulse noise, have been identified as particularly problematic in the reverse channel.

2.3.1 Ingress Noise

Ingress is the entrance of ambient external signals into a network. Although HFC systems are theoretically closed, meaning that signals both within and external to a network remain so, the shielding effectiveness of plants is often compromised by leaky hardware. As a result, external signals in frequency bands that coincide with the reverse channel bandwidth can enter HFC networks and cause interference. Examples of potential reverse channel ingress signals include amateur radio, citizen's band, and short-wave radio transmissions, as shown in Figure 2.1.

Studies have revealed that ingress may enter HFC networks in a variety of ways. First, coaxial connectors are generally attached to cables by crimping. If shield connections are improperly crimped, the shield can act as a receiving antenna. [2] Furthermore, as connections wear, shields may break at the crimps, thus compromising a system's immunity to ingress. A second potential cause of ingress is illegal hookups to cable networks. People making unauthorized connections to a network may not understand the importance of shielding and proper connections. In addition, cable operators have found illegal hookups in which twin-lead cable has been used instead of coaxial cable [2], which results not only in potential points of entry for ingress, but also in impedance mismatches, which cause ISI as described previously. Older drop

cables, particularly those installed prior to 1980, may not be as well shielded as newer cables, and they may be less effective at preventing ambient signals from entering the network. Television sets also contribute to ingress problems because they are, by necessity, broadband devices that receive over-the-air signals. Finally, common accessories such as splitters, video games, video cassette recorders, and their additional required cabling also offer entry points for unwanted radio-frequency signals.

Because nearly all potential points of entry for ingress are located in subscriber premises, cable operators have little or no control over ingress noise. The problem is compounded in the reverse channel by what is known as the "funneling effect," which is the accumulation of all ingress signals at the headend. Hence, ingress caused by a single subscriber's equipment affects reverse channel communications by *every* subscriber because the noise seen by the headend receiver contains a component that is the sum of all ingress noise in the reverse channel. In contrast, ingress in the downstream bandwidth affects only those subscribers downstream of the ingress point of entry. Often, ingress may affect only a single subscriber. Therefore, on average, many fewer subscribers are affected by ingress, and the problem is not considered especially significant in the downstream direction.

A recent CableLabs study found that roughly 5% of subscriber premises provide less than 36 dB shielding. [10] These subscribers are likely to contribute ingress noise to the reverse channel. If the precise characteristics of ingress as a function of time could be predetermined, then a reverse channel transmission scheme could be designed to operate in the bandwidth not affected by ingress. However, the bands allocated in the over-the-air spectrum for potential ingress signals overlap almost entirely the reverse channel allocation in the HFC spectrum, as shown in Figure 2.1. Furthermore, the bandwidths of individual interfering signals can vary from fewer than 100 Hz to several megahertz, depending on the ingress source. [11] In addition, the durations, spectral locations, and power levels of interferers can vary. Consequently, only general comments can be made regarding the characteristics of ingress noise in the reverse channel. For example, Rogers Cable Systems reports in a recent study that ingress in the band from 5 to 20 MHz is typically much more severe than ingress in the band above 20 MHz. [12] Because ingress is a severe problem that is almost certain to affect HFC reverse channels, an effective digital transmission scheme must be able to maintain a desired performance level in the presence of ingress noise.

2.3.2 Impulse Noise

The second noise source identified as severe in the reverse channel is *impulse noise*. In contrast to ingress noise, which consists of one or more disturbers that are localized in frequency but potentially high-power, impulse noise is broadband. Furthermore, whereas ingress signals can interfere with transmission for a significant span of time, impulse noise is a sporadic, short-duration disturber. In general, the spectral energy density of impulse noise in HFC networks decreases with increasing frequency. Consequently, the reverse channel is affected by more severe impulse noise than are the downstream channels. The causes of impulse noise in HFC networks are not well understood, but several potential sources have been identified, including power line discharges, lightning, industrial machinery, and home appliances. [12] [13] Impulse noise enters HFC networks via the same mechanisms as does ingress noise. In particular, faulty connectors and insufficient grounding of the cable shield can allow impulse noise to enter the HFC plant.

2.4 Channel Model

From the discussion in the previous section, it is clear that "typical" HFC reverse channels suffer from a variety of impairments, most of which are difficult or impossible to characterize precisely. The reverse channel frequency spectrum can be measured to determine either an instantaneous or average channel profile, but neither is likely to represent the reverse channel accurately enough to be meaningful. Likewise, modeling the channel and, particularly, reverse channel noise for computer simulations is difficult. However, based on the preceding discussions, it is possible to construct a channel model representative of the worst-case reverse channel transmission environment. By showing that a particular transmission scheme achieves a desired performance on the worst-case channel, acceptable performance can be assured on less severe, more common channels.

A useful HFC reverse channel model should capture effects of the four degradations described in this chapter:

- Thermal noise;
- Micro-reflections;

- Ingress noise;
- Impulse noise.

First, because thermal noise is frequency-independent, it is modeled as additive white Gaussian noise (AWGN). Second, as described in the previous section, micro-reflections are the result of mismatched impedances on the network, and they cause deviations and nulls in a channel's frequency response. Hence, the effects of microreflections on the channel frequency response can be simulated by constructing a network model that includes a variety of impedance mismatches. Next, ingress noise is composed of relatively narrow-bandwidth disturbers that could be modeled as regions in the noise spectrum with higher power spectral densities than the ambient AWGN. However, the effect of ingress on the reverse channel signal-to-noise ratio (SNR) suggests a simpler model for ingress noise, particularly because this dissertation proposes Discrete Multi-Tone modulation for reverse channel transmission. To illustrate, assume that a channel with a flat frequency response is corrupted by ingress noise and AWGN. Assuming a uniform transmit power distribution, the channel SNR as a function of frequency is computed using

$$\mathrm{SNR}(f) = \frac{PH(f)}{S_N(f)}, \tag{2.2}$$

where, P is the transmit power per Hertz, $H(f)$ is the channel insertion loss, and $S_N(f)$ is the power spectral density of the channel noise. If the channel SNR computed from Equation 2.2 is plotted as a function of frequency, regions of the frequency band corrupted by ingress noise appear as nulls or minima, as shown in Figure 2.7. Therefore, the effects of ingress noise on the SNR are essentially the same as the effects of impedance mismatches, and, when Discrete Multi-Tone is chosen as the transmission technique, demonstrating adequate system performance on a channel exhibiting spectral nulls implies that the system will also perform well in the presence of ingress noise. The last impairment, impulse noise, can be modeled as a transient increase in the channel AWGN level. Hence, the AWGN power can be increased to simulate impulse noise.

To generate a channel model that includes the channel degradations described,

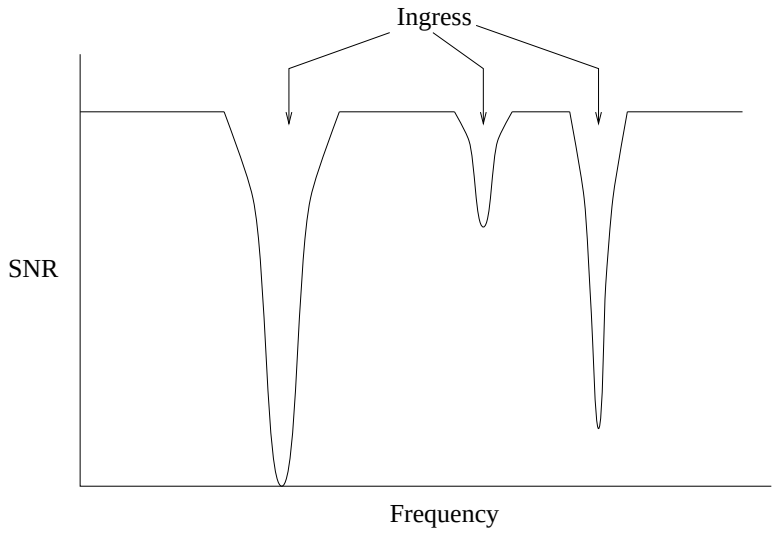

Figure 2.7: Effect of ingress noise on channel SNR

a reverse channel frequency response corresponding to the simple network configuration shown in Figure 2.8 was computed.[1] Because the majority of degradations to reverse channel signals are introduced at points between the fiber node and subscriber premises, only the coaxial portion of the HFC network has been modeled. Distribution and line extender amplifiers are not included in the model. Real HFC systems may include amplifiers in the reverse channel; if so, then the levels of signals transmitted at higher frequencies would be increased. Hence, the model yields a worst-case frequency response. The model network consists of 0.5-inch copperclad coaxial cable that delivers signals through taps to as many as fifteen subscribers. The attenuation characteristics of various lengths of this particular 0.5-inch coaxial cable were shown in Figure 2.3. The subscriber premise for which the frequency response was computed is denoted with a dashed box in Figure 2.8. In the model configuration, both the source impedance at the home considered and the headend termination impedance are assumed to be 75Ω, but it is assumed that the drop cables in some homes are either unterminated (that is, open-circuited) or improperly terminated.

[1]Thanks to Professor D. G. Messerschmitt of the University of California at Berkeley for making his transmission line modeling program available to our research group.

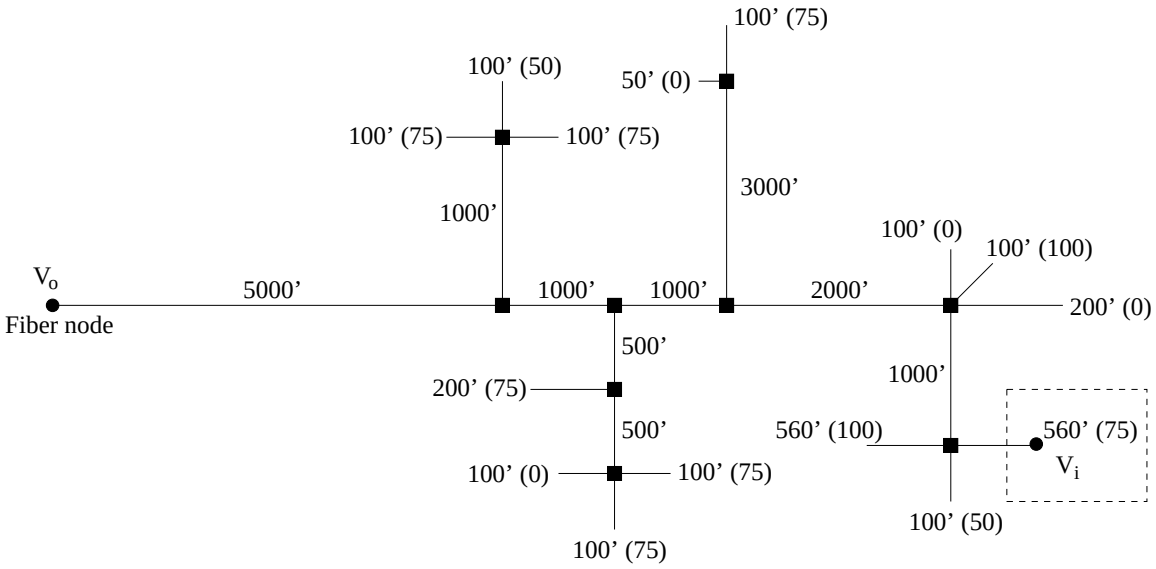

Figure 2.8: Simulated system configuration

The values assumed for the remote terminal source impedances are shown in parentheses in Figure 2.8. The line lengths were chosen so that the maximum length of coaxial cable between any remote terminal and the fiber node is two miles.

Figure 2.9 shows the frequency response magnitude of the chosen reverse channel. The variation in frequency response magnitude caused by the impedance mismatches is evident in the figure. Also shown clearly is the frequency-dependence of the response magnitude due to the coaxial cable characteristics and absence of amplifiers.

Even though the network model technically includes only the effects of impedance mismatches, as described previously, when the reverse channel SNR is considered, the nulls could just as well have been caused by ingress noise. Hence, the reverse channel shown in Figure 2.9 can be considered the *equivalent channel*, meaning the frequency response includes the effects of both the channel and all noise except AWGN and transient noises such as impulse noise. The addition of both AWGN and impulse noise is straightforward when Discrete Multi-Tone modulation is simulated, as will be shown in Chapter 3. Figure 2.10 shows a frequency-dependent SNR that might be achieved in the band from 24.336–28.752 MHz when a flat transmit-power-to-AWGN level of 75.0 dB is assumed.

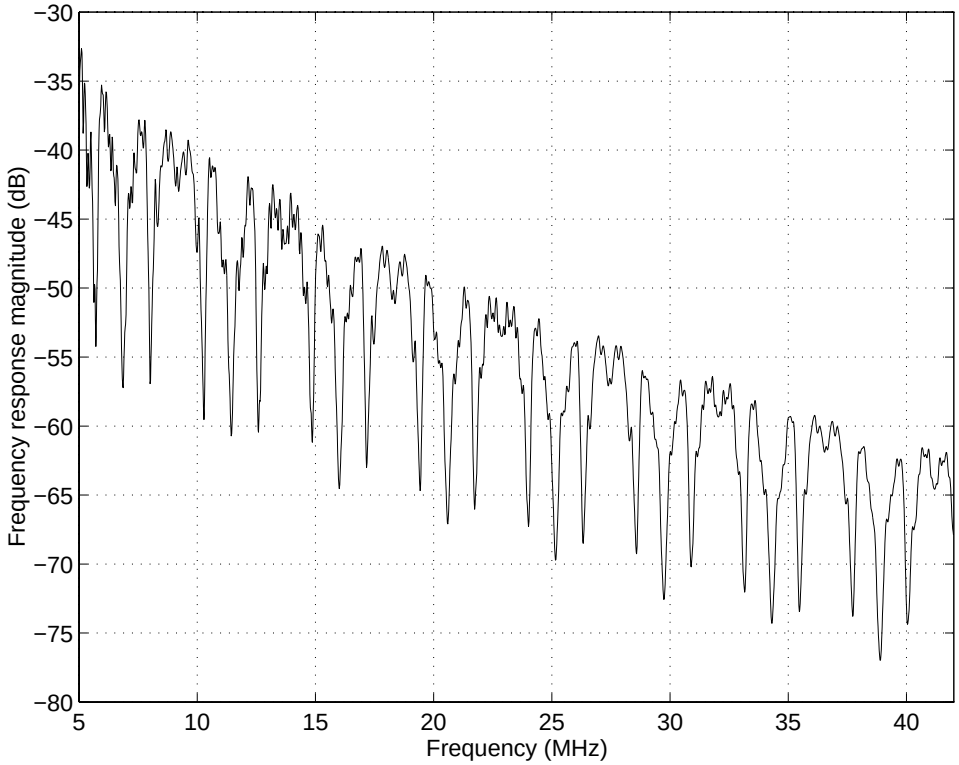

Figure 2.9: Frequency response magnitude of simulated return channel

2.5 Summary

In this chapter, the HFC network architecture was described, and the limitations imposed by the architecture on reverse channel transmission were identified. In particular, the inability of remote terminals to receive signals in the reverse channel bandwidth due to HFC hardware was noted. This constraint will drive the development of reverse channel protocols for multicarrier modulation in Chapter 5. Also in this chapter, various reverse channel transmission impairments and their causes were described, including micro-reflections, thermal noise, ingress noise, and impulse noise. These impairments were then incorporated into a worst-case equivalent reverse channel model, which will be used in Chapter 3 to evaluate Discrete Multi-Tone as a solution for reverse channel transmission.

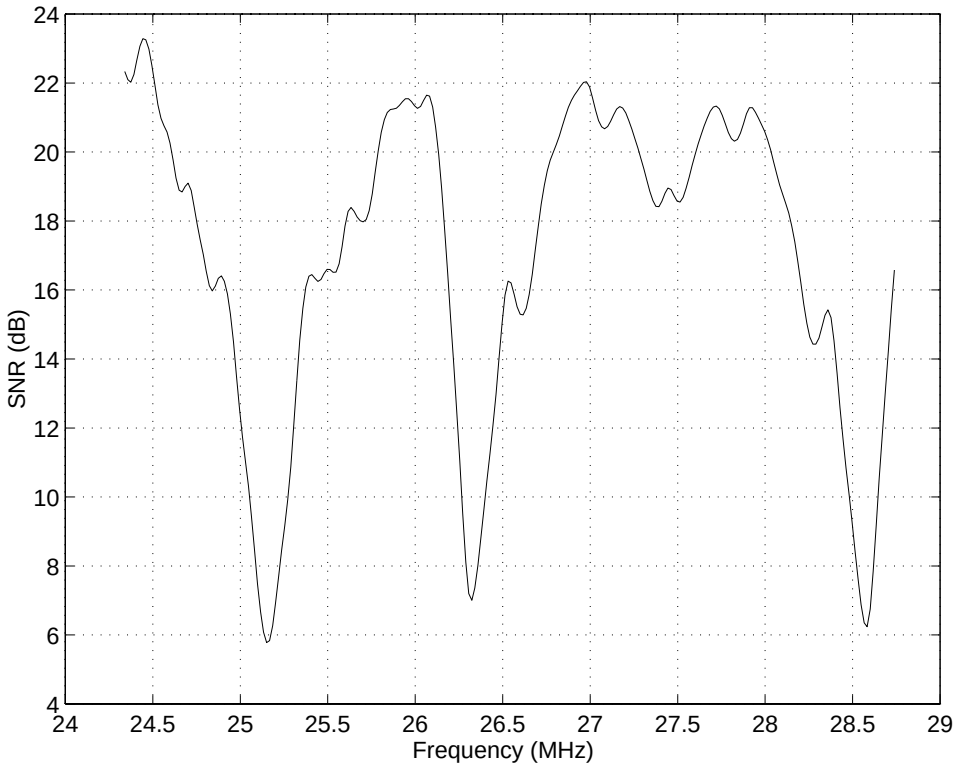

Figure 2.10: Possible SNR in band from 24.336–28.752 MHz

Chapter 3

Discrete Multi-Tone Modulation for Upstream HFC Transmission

Although numerous techniques may be used to transmit digital signals on bandlimited channels, given a finite-complexity constraint and the need to support a high bit rate with low latency, there are two practical options: single-carrier quadrature amplitude modulation (QAM) with equalization and multicarrier modulation. In single-carrier QAM, symbols are transmitted using the entire channel bandwidth. The symbols are then decoded one by one in the receiver. Because practical channels cause intersymbol interference (ISI), an equalizer is used to reduce the interference between successively transmitted symbols, thereby improving the performance of the system while maintaining a fixed complexity. Equalization is, however, an inherently suboptimal detection method for practical channels. Furthermore, finite-length equalizers may not perform well on channels that have significant deviations in their frequency response magnitudes. In contrast, multicarrier modulation is well-suited for these types of channels because it partitions a channel into a set of orthogonal subchannels, each of which supports a distinct carrier. Discrete Multi-Tone (DMT) modulation is a specific type of multicarrier modulation that achieves excellent performance with finite complexity. Hence, because it offers numerous advantages, DMT is proposed as a solution for upstream transmission on HFC networks.

The chapter begins with a review of DMT in Section 3.1. In particular, the purpose of the cyclic prefix is examined, and the calculation of the aggregate bit rate a channel can support is described. Section 3.2 discusses the optimality of DMT

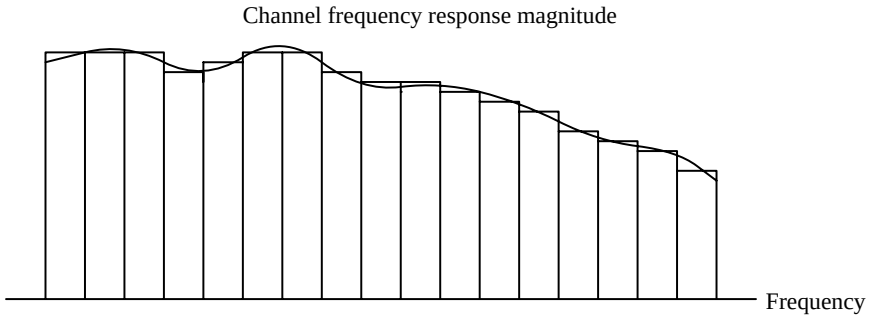

Figure 3.1: A channel partitioned into subchannels

for HFC reverse channels. An example shows that DMT automatically optimizes the bandwidth used to transmit data. In Section 3.3, the achievable performance of DMT on the example channel simulated in Chapter 2 under various power and noise circumstances is then quantified. Based on these results, a reference configuration for DMT-based reverse channel transmission is presented in Section 3.4.

3.1 Discrete Multi-Tone Modulation

This section briefly reviews Discrete Multi-Tone modulation. For additional details, the reader is referred to [14] through [19].

Discrete Multi-Tone (DMT) modulation uses an *inverse discrete Fourier transform (IDFT)* to partition a transmission channel into a set of orthogonal, equal-bandwidth subchannels so that the frequency response of each subchannel is roughly constant across its bandwidth, as illustrated in Figure 3.1. The resulting subchannels are almost memoryless, and the small amount of intersymbol and intersubchannel interference caused by the channel's non-unity impulse response length can be eliminated by use of a *cyclic prefix*. As its name implies, the cyclic prefix precedes each symbol, as shown in Figure 3.2. It is a copy of the last ν time-domain (that is, post-IDFT) samples of each DMT symbol. Hence, the cyclic prefix carries redundant information and is transmission overhead. If there are no more than $\nu + 1$ non-zero samples in the channel impulse response, then ISI caused by each symbol is confined to the cyclic prefix of the following symbol. By discarding the cyclic prefix samples in the receiver, ISI is eliminated completely, as illustrated by the following example: Assume the channel

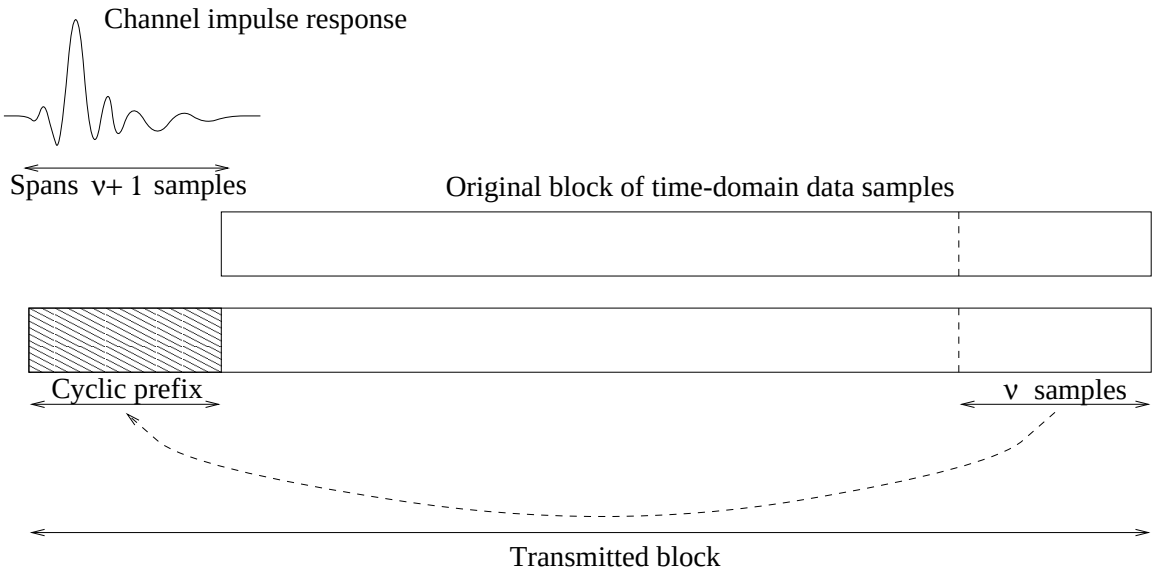

Figure 3.2: Illustration of cyclic prefix

impulse response length is 4 samples, and the cyclic prefix spans 3 samples, which is the minimum length required to eliminate ISI in this case. Assume further that the IDFT size is 8, meaning each transmitted DMT symbol contains 8 independent (time-domain) data samples. Denote the current block of time-domain data samples as $\mathbf{x}$, the previous block as $\tilde{\mathbf{x}}$, the channel impulse response as $\mathbf{h}$, and the current received time-domain block as $\mathbf{y}$. Assuming infinite-resolution analog-to-digital and digital-to-analog converters and neglecting noise, the current received discrete-time signal can be written as $y_k = x_k * h_k$, where $*$ denotes linear convolution. Under the impulse response length and cyclic prefix length assumptions, the output samples y_k are

$$\begin{aligned} y_k &= x_k * h_k \\ &= [\cdots\ \tilde{x}_5\ \tilde{x}_6\ \tilde{x}_7\ x_5\ x_6\ x_7\ x_0\ x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7\ \cdots] * [h_0\ h_1\ h_2\ h_3]. \end{aligned}$$

Denoting the received cyclic prefix samples as y_{-3}, y_{-2}, and y_{-1}, the entire received

block corresponding to data block $\mathbf{x}$ and its cyclic prefix can be written as

$$\begin{bmatrix} y_{-3} \\ y_{-2} \\ y_{-1} \\ y_0 \\ y_1 \\ \vdots \\ y_7 \end{bmatrix} = \begin{bmatrix} h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 & 0 & h_0 & 0 & 0 \\ 0 & h_3 & h_2 & 0 & 0 & 0 & 0 & 0 & h_1 & h_0 & 0 \\ 0 & 0 & h_3 & 0 & 0 & 0 & 0 & 0 & h_2 & h_1 & h_0 \\ 0 & 0 & 0 & h_0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 \\ 0 & 0 & 0 & h_1 & h_0 & 0 & 0 & 0 & 0 & h_3 & h_2 \\ 0 & 0 & 0 & h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 & h_3 \\ 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} \tilde{x}_5 \\ \tilde{x}_6 \\ \tilde{x}_7 \\ x_0 \\ \vdots \\ x_7 \end{bmatrix}.$$

An examination of the set y_k reveals that although the received cyclic prefix samples are corrupted by samples of $\tilde{\mathbf{x}}$, the data samples y_0 through y_7 contain contributions only from $\mathbf{x}$. Hence, the cyclic prefix can be discarded in the receiver prior to reconstructing the data stream using a *discrete Fourier transform (DFT)*, and, if the cyclic prefix is long enough, ISI is eliminated completely. Therefore, a traditional equalizer is not required in the DMT receiver. If the cyclic prefix is not long enough, the received signal incurs some distortion. This effect is discussed and quantified in Appendix B.

In the DMT transmitter, a bitstream is encoded as a set of quadrature amplitude modulated (QAM) subsymbols, where each QAM subsymbol represents a number of bits determined by the signal-to-noise ratio (SNR) of its associated subchannel. The following derivation yields an equation that can be used to compute the number of bits each subchannel supports. The derivation assumes that a QAM signal is transmitted on an ISI-free additive white Gaussian noise channel. Given that the well-known Q-function is defined by

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{\frac{y^2}{2}} dy,$$

the probability that the receiver incorrectly decodes a transmitted QAM symbol is

$$P_e = N_e \, Q\left[\frac{d_{\min}}{2\sigma}\right], \tag{3.1}$$

where N_e is the average number of nearest neighbors in the QAM constellation, σ^2 is the received noise variance per dimension, and $d_{\min}$ is the *minimum distance* between points in the QAM constellation at the channel output. In terms of the distance d between points in the input constellation,

$$d_{\min}^2 = d^2 \ |H|^2,$$

where $|H|$ is the channel gain. For QAM constellations, $N_e \leq 4$, and Equation 3.1 is consequently written as

$$P_e \leq 4 \ Q \left[\frac{d_{\min}}{2\sigma} \right]. \tag{3.2}$$

At the receiver, the SNR is

$$\text{SNR} = \frac{\mathcal{E}|H|^2}{2\sigma^2}, \tag{3.3}$$

where $\mathcal{E}$ is the average input signal energy per two-dimensional QAM symbol. If all points in the QAM constellation are equally likely to be transmitted, then

$$\mathcal{E} = \frac{d^2(2^b - 1)}{6}, \tag{3.4}$$

where b is the number of bits the constellation represents. Substituting Equation 3.4 into Equation 3.3,

$$\text{SNR} = \frac{d^2(2^b - 1)|H|^2}{12\sigma^2}. \tag{3.5}$$

Defining the *SNR gap* $\Gamma(P_e)$ as

$$\Gamma(P_e) = \frac{d^2|H|^2}{12\sigma^2} = \frac{d_{\min}^2}{12\sigma^2}, \tag{3.6}$$

a manipulation of Equation 3.5 yields an approximation of the number of QAM bits a channel can support:

$$b = \log_2 \left(\frac{\text{SNR}}{\Gamma(P_e)} + 1 \right). \tag{3.7}$$

The SNR gap $\Gamma(P_e)$ is an estimate of the difference between Shannon's channel capacity and the achievable rate of a practical system. Note that, from Equation 3.1, the SNR gap is computed as

$$\Gamma(P_e) = \frac{[Q^{-1}(\frac{P_e}{N_e})]^2}{3}. \tag{3.8}$$

Using Equation 3.8 and assuming $N_e = 4$, $\Gamma(10^{-7}) = 9.8$ dB, and $\Gamma(10^{-9}) = 11.1$ dB.

Returning to a set of DMT subchannels, given a desired symbol error probability on each subchannel, an expression for the number of bits that can be supported by the ith subchannel is

$$b_i = \log_2 \left(\frac{\mathrm{SNR}_i}{\Gamma(P_e)} + 1 \right), \tag{3.9}$$

where b_i is the number of bits supported by the ith subchannel, SNR_i is the signal-to-noise ratio of the ith subchannel, and $\Gamma(P_e)$ is the SNR gap at the desired symbol error probability as defined previously. Similar to before,

$$\mathrm{SNR}_i = \frac{\mathcal{E}_i |H_i|^2}{2\sigma_i^2},$$

where $\mathcal{E}_i$ is the energy allocated to the ith subchannel, and $|H_i|$ and σ_i^2 are the gain and noise variance, respectively, of subchannel i. Using Equation 3.9, SNRs required to support various QAM constellation sizes with designated symbol error probabilities can be computed. Table 3.1 gives the subchannel SNRs required to support 2 through 8 QAM bits with $P_e = 10^{-7}$ and 10^{-9}.

As a simple DMT bit allocation example, assume a transmission channel is divided into 4 subchannels[1] with SNRs of 15.5, 22.5, 19.0, and 2.0 dB, respectively. Then, according to Table 3.1, to achieve an (uncoded) received symbol error probability of 10^{-7}, the subchannels can support 2, 4, 3, and 0 bits, respectively. Hence, DMT can transmit up to 9 bits per symbol[2] in this configuration, with the first three subchannels supporting 4-QAM (or, equivalently, QPSK), 16-QAM, and 8-cross QAM. Because

[1] In reality, four subchannels would seldom be sufficient to satisfy the requirement of flat frequency response magnitude over the bandwidth of each subchannel.

[2] In some cases it may be desirable to allocate fewer bits per subchannel than the channel can support.

Table 3.1: SNRs required to support various numbers of QAM bits at $P_e = 10^{-7}$ and $P_e = 10^{-9}$

	Required SNR (dB)	
Bits	$P_e = 10^{-7}$	$P_e = 10^{-9}$
2	14.6	15.9
3	18.3	19.6
4	21.6	22.9
5	24.7	26.0
6	27.8	29.1
7	30.8	32.1
8	33.9	35.2

the SNR of the last subchannel is insufficient to support even one bit at 10^{-7} symbol error probability, it is not used. The total bit rate of the system when 9 bits are allocated per symbol is then (9 bits/symbol)(W symbols/s), where W is the nominal bandwidth of each subchannel. Note that W is also the symbol rate of the DMT system.

During each DMT symbol period, a set of subsymbols is input as a block to a complex-to-real IDFT. The cyclic prefix is added to the output, and the result is converted from digital to analog format. If passband transmission is required, the resulting signal, a sum of real time-domain sinusoids, is then upconverted to the desired carrier frequency and transmitted over the channel. In the receiver, after downconversion and analog-to-digital conversion, the cyclic prefix is stripped, and the samples are input to a DFT. Each output value is then scaled by a single complex number to compensate for the magnitude and phase of each subchannel's frequency response, and a memoryless detector decodes the resulting symbols. The set of complex multipliers is known as the *frequency-domain equalizer (FEQ)*. Changes in the channel magnitude or phase are accommodated by updating the FEQ taps as the system operates. A generalized passband DMT transmitter and receiver pair is shown in Figure 3.3.

During steady-state operation, the subchannel SNRs are monitored in a data driven manner by the receiver, and the bit distribution is modified as necessary at the transmitter to maintain a desired system performance. This adaptivity enables DMT to alleviate problems caused by changes in the channel frequency response or

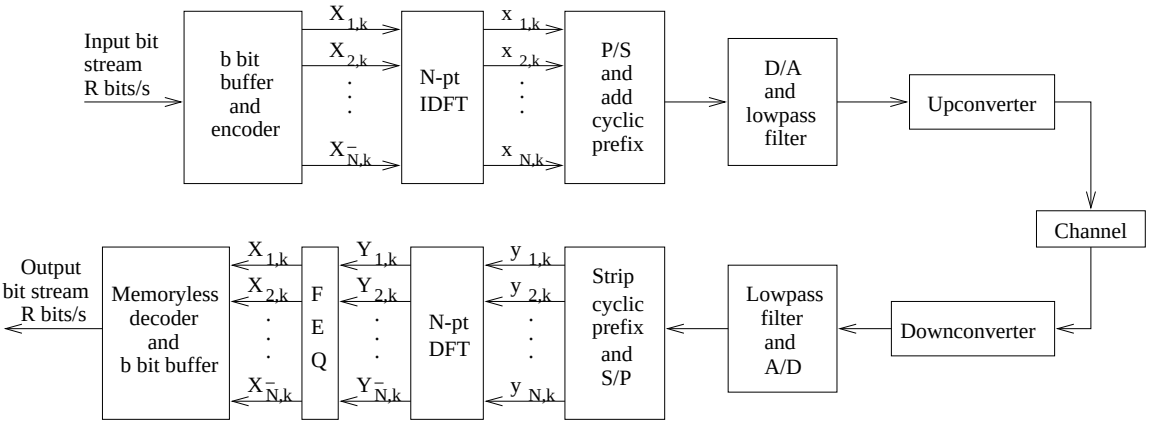

Figure 3.3: Passband DMT block diagram

noise while a system operates. If the attenuation or noise in a subchannel becomes severe enough to compromise the system error performance, fewer or no bits are assigned to that subchannel while the noise or attenuation persists. Upon detecting a degradation in a subchannel SNR, the receiver computes a modified bit distribution that better achieves the desired error performance. Depending on the SNR of a degraded subchannel, some or all of its bits may be moved via a *bit swap* algorithm to one or more other subchannels that can support additional bits. The bit distribution change is reported to the transmitter, where it is implemented. Therefore, even unpredictable, bursty, high-power noise such as ingress from amateur radio or other radio-frequency interferers can be tolerated by a DMT system, assuming that either the resulting channel can continue to support the required data rate or the data rate can be decreased and that the interference does not overload the receiver and cause nonlinear operation. Dramatic changes in the channel itself can also be accommodated by an adaptive DMT system, provided that the adaptation speed of the bit swap algorithm is sufficient to track the changes. Note that changes in the channel frequency response or noise during operation that actually improve any of the subchannel SNRs can be used to increase the system's overall data rate, if desired. Alternatively, if a fixed data rate is desired, improvements in subchannel SNRs yield higher noise immunity by increasing the *noise margin*. The noise margin is the SNR in excess of what is required to support a particular number of bits on a subchannel. For example, if a subchannel SNR is 28.0 dB, and a symbol error probability of 10^{-7}

is desired, then imposing a noise margin of 6.0 dB means only 4 bits are assigned to that subchannel, even though the SNR is sufficient to support 6 bits. Operating with a noise margin of γ_n dB on each subchannel implies that the subchannel noises can increase by γ_n dB before the symbol error probability is compromised.

Depending on the system symbol rate, noise margin, and noise power, impulse noise may or may not affect the performance of a DMT system. With respect to single-carrier systems, multicarrier systems offer improved impulse noise immunity. Because the DMT receiver processes received samples in blocks, after the receiver transform the energy of any impulse noise that corrupted some of the received samples is spread evenly among the subchannels. Hence, impulse noise causes a temporary decrease in each subchannel SNR. If the subchannel noise margins are large enough, impulse noise may not affect the system error performance. If, however, the expected impulse noise is severe enough to exceed the noise margin, then forward error correction with interleaving can be applied to mitigate errors. [20]

3.2 Optimality of DMT

One advantage of DMT is that it automatically optimizes the channel bandwidth that is used for data transmission. Regions of the transmit spectrum that cannot support meaningful communications are avoided when the bit distribution of a channel is computed. Practical single-carrier transceivers are much less proficient at optimizing the transmission bandwidth: most finite-length equalizers cannot produce required nulls and notches in the frequency spectrum as accurately as DMT can simply by turning off subchannels. A simple example illustrates the automatic bandwidth optimization by DMT.

Figure 3.4 shows the insertion loss of two lengths, 1 mile and 2 miles, of 0.5-inch coaxial cable as a function of reverse channel frequency. Assuming the only channel noise is additive white Gaussian noise, if a uniform power distribution is chosen so that the transmit power-to-noise ratio is 50 dB, Figure 3.5 illustrates the resulting reverse channel SNRs as functions of frequency. Assuming the reverse channel is partitioned into 1024 subchannels and a 6.0 dB noise margin is imposed, Figure 3.6 shows the DMT bit distributions computed using Equation 3.9 for the two line lengths. In computing the bit distributions, it was assumed that subchannels with SNRs insufficient

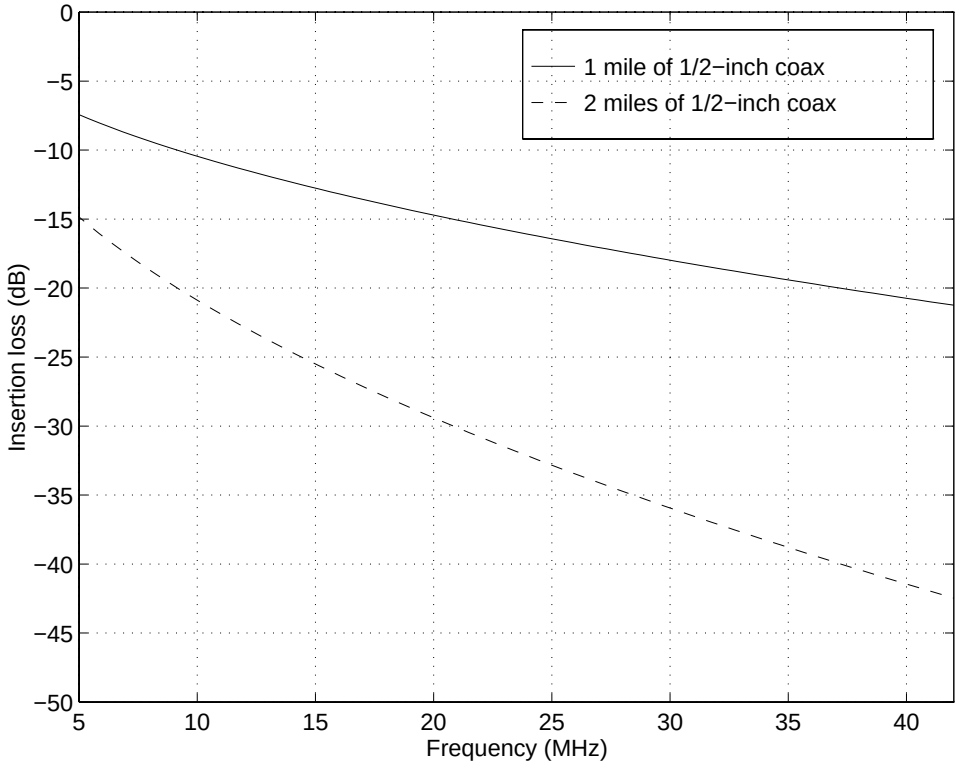

Figure 3.4: Insertion losses across the reverse channel bandwidth of 0.5-inch coaxial cable

to support at least two bits are not used.

As Figure 3.6 illustrates, the 1-mile coaxial cable is capable, assuming the previously specified transmit power, noise, and noise margin values, of supporting data using the entire reverse channel bandwidth. Up to 9 bits are supported near the lower edge of the band, and at least 5 bits can be supported over the entire bandwidth. In contrast, under the same assumptions, the attenuation of the 2-mile cable is severe enough that the line can support two or more bits only in the lower half of the reverse channel bandwidth. Hence, subchannels above roughly 19 MHz are simply not used. This result reveals a significant advantage of DMT: a single design can be used on a wide range of channels because the transmit bandwidth is optimized for each channel. Additional details on bandwidth optimization can be found in [21].

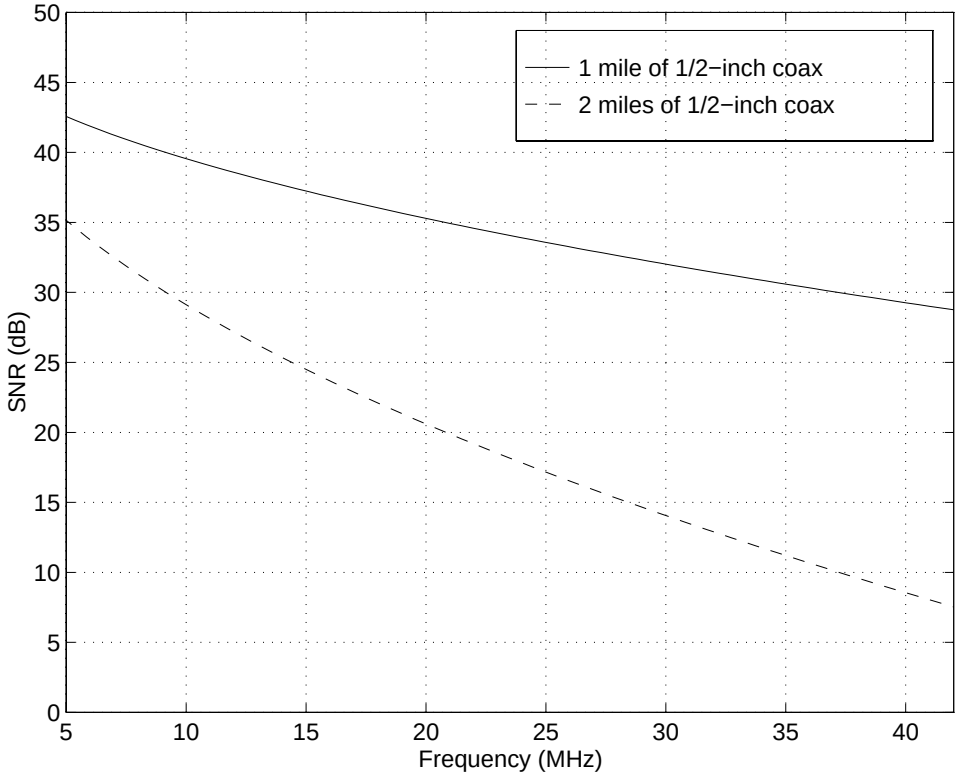

Figure 3.5: Reverse channel SNR

3.3 Achievable Performance of DMT in HFC Reverse Channels

Given the advantages of DMT for HFC reverse channel transmission— immunity to ingress, channel nulls, and impulse noise, and the ability to adapt to changes in the channel characteristics—the next step is to quantify the achievable performance of DMT on the worst-case channel modeled in Chapter 2. Neglecting for the time being the multi-access nature of the reverse channel, the example channel from Chapter 2 is assumed to be the equivalent channel from a particular remote terminal to the headend, meaning that its frequency response includes the effects of both channel variation and all noise except AWGN and impulse noise. The transmit power distribution is assumed to be uniform in frequency, and no analog equalization is assumed; consequently, the SNR of the reverse channel tends to decrease with frequency due

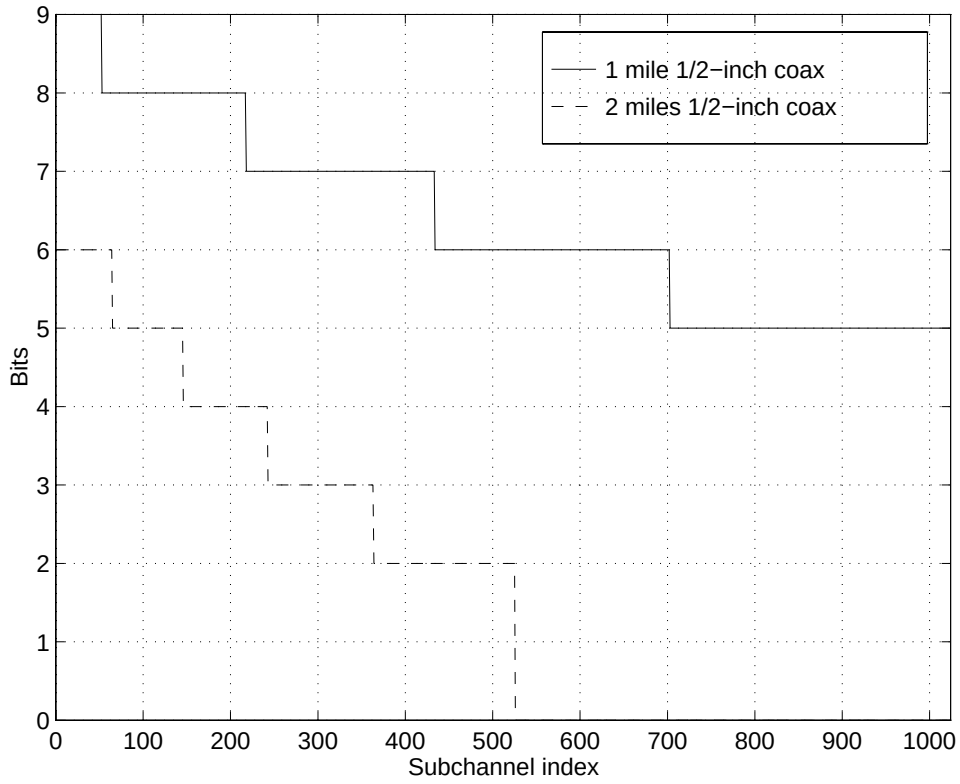

Figure 3.6: Bit distributions: 1024 subchannels, 6 dB noise margin

to the coaxial cable loss.

To achieve a desired symbol error probability of 10^{-7} and assuming subchannels unable to support at least two bits are not used, Figure 3.7 shows the DMT bit distribution when (almost all of) the reverse channel is partitioned into 1024 34.5-kHz-wide subchannels, each with 0 dB noise margin. Hence, the sampling rate is 70.656 MHz, and the symbol rate is 32 kHz with a cyclic prefix length of 160 samples. The ratio of transmit power to AWGN used in the simulation was 80 dB, which, along with the equivalent channel insertion loss, yields the plot of SNR as a function of frequency that is shown in the lower half of Figure 3.7. Using this 1024-subchannel configuration, the aggregate achievable data rate in the example reverse channel is 168.7 Mbps. Of course, assuming the equivalent channel remains constant, the aggregate upstream data rate is a function of the transmit power and AWGN noise levels. Table 3.2 shows the achievable data rates and their corresponding spectral efficiencies in bits/s/Hz for

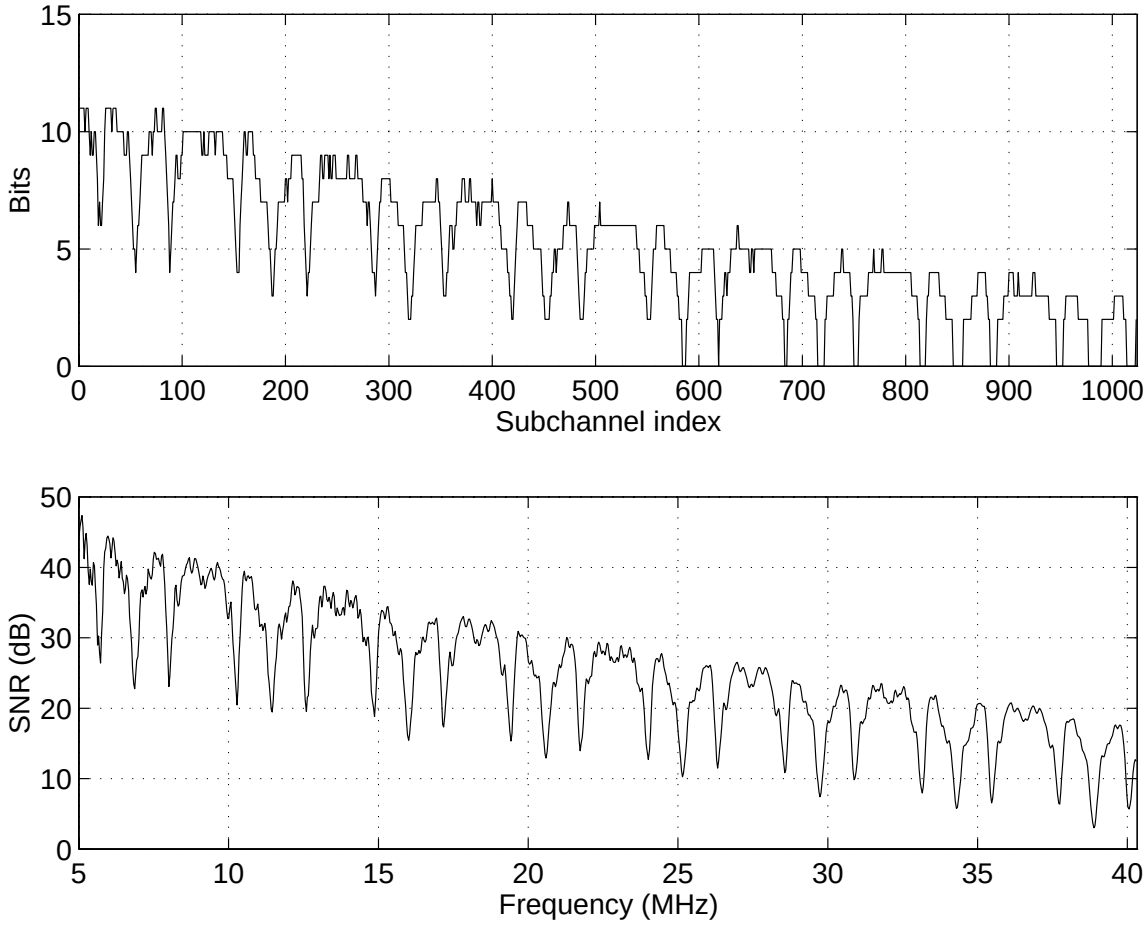

Figure 3.7: Example reverse channel SNR and corresponding bit distribution with an 80 dB transmit power-to-AWGN ratio ($P_e = 10^{-7}$)

various transmit-power-to-AWGN ratios.

As mentioned in Chapter 2, the effect of impulse noise can be considered a temporary increase in the AWGN level, or, equivalently, a decrease in the power-to-AWGN ratio. Hence, rates at the lower power-to-AWGN values given in Table 3.2 might represent a lower bound on the instantaneous performance of the system during those symbol periods affected by impulse noise, while rates corresponding to the higher power-to-AWGN ratios are more representative of the performance during "normal" operation, assuming no noise margin. As discussed previously, imposing a noise margin may prevent degradations due to some types of impulse noise.

Table 3.2: Achievable aggregate data rates and spectral efficiencies on example reverse channel ($P_e = 10^{-7}$)

Power-to-AWGN ratio (dB)	Aggregate data rate (Mbps)	Spectral efficiency (bits/s/Hz)
71	90.0	2.55
74	109.5	3.10
77	138.2	3.91
80	168.7	4.78
83	199.1	5.64
86	228.3	6.46
89	255.8	7.24
92	281.0	7.95

Table 3.3: Achievable aggregate data rates and spectral efficiencies on example reverse channel ($P_e = 10^{-9}$)

Power-to-AWGN ratio (dB)	Aggregate data rate (Mbps)	Spectral efficiency (bits/s/Hz)
71	83.8	2.37
74	99.7	2.82
77	126.0	3.57
80	155.8	4.41
83	185.9	5.26
86	215.4	6.10
89	243.4	6.89
92	270.0	7.64

If the required decoded symbol error probability is 10^{-9} rather than 10^{-7}, which might be the case if error correcting codes are not used in conjunction with DMT, then the aggregate bit rates achievable on the example channel decrease somewhat. Maintaining the other assumptions from the previous simulations, the bit rates with $P_e = 10^{-9}$ are given in Table 3.3. Also shown are the associated spectral efficiencies in bits/s/Hz. Comparing Tables 3.2 and 3.3, the loss in achievable rate over the entire reverse channel when the required symbol error probability is decreased from 10^{-7} to 10^{-9} ranges from just over 6 Mbps with 71 dB transmit-power-to-AWGN ratio to 11 Mbps with 92 dB transmit-power-to-AWGN ratio.

The previous simulations assumed, rather arbitrarily, that the reverse channel is

partitioned into 1024 subchannels, each of which is 34.5 kHz wide. In general, the reverse channel can be partitioned into any convenient number of subchannels, as long as the condition of nearly-flat frequency responses over the subchannel bandwidths is met. However, one cost-effective implementation of DMT uses the fast Fourier transform (FFT) to compute efficiently the DFT and IDFT required in the receiver and transmitter, respectively. To exploit the computational efficiency offered by the FFT, the number of subchannels should be an integer power of two. Cost and performance constraints then dictate the precise number of subchannels into which a channel is partitioned. Assuming a fixed total channel bandwidth, as the number of subchannels increases, so does the DMT symbol period. Because the required cyclic prefix is a function of the duration of the channel impulse response, to eliminate ISI it must span the same amount of time regardless of how many subchannels are used. Hence, the relative overhead required to transmit the cyclic prefix decreases as the number of subchannels and, consequently, the DMT symbol period increase. In other words, as the number of subchannels increases, the percentage of time the channel is used to transmit the redundant cyclic prefix samples decreases, and the overall system efficiency increases. The penalty for this increase in efficiency with increased numbers of subchannels is, of course, an increase in system complexity resulting from the increase in FFT size.

Given the discussion of microreflections in Chapter 2, a cyclic prefix of approximately 2 μs duration should be sufficient to eliminate nearly all ISI between successively transmitted DMT symbols in HFC reverse channels. Assuming (for convenience and reasons that will be discussed in Appendix B) that a 2.26 μs cyclic prefix is used, which corresponds to 160 samples when the sampling rate is 70.656 MHz (approximately twice the available reverse channel bandwidth), Table 3.4 shows the achievable aggregate reverse channel bit rates for various numbers of subchannels and transmit-power-to-AWGN ratios when $P_e = 10^{-7}$. Also shown are the computational complexities, in millions of instructions per second (MIPS), required to compute the FFT required in the DMT transmitter or receiver for each FFT size. The FFT is the most computationally intensive operation performed by the transmitter and receiver, and therefore the number of MIPS required to compute the FFT is representative of the remote terminal complexities. Counting a multiply-accumulate as a single instruction, the number of instructions required to compute the N-point FFT of a real

Table 3.4: Achievable aggregate data rates (in Mbps) on example reverse channel with $P_e = 10^{-7}$ as a function of FFT size

Power-to-AWGN ratio (dB)	FFT size ($N = 2\bar{N}$)					
	8192	4196	2048	1024	512	256
71	95.1	93.5	90.0	83.8	74.2	59.6
74	115.9	113.7	109.5	102.4	90.4	71.8
77	146.1	143.4	138.2	128.7	114.2	90.7
80	178.4	175.1	168.7	157.2	139.0	111.9
83	210.4	206.3	199.1	185.6	163.9	131.8
86	241.4	236.7	228.3	213.4	187.2	151.5
89	270.2	265.2	255.8	238.8	210.3	169.5
92	296.9	291.3	281.0	262.1	231.2	186.3
Complexity (MIPS)	1822	1638	1442	1222	969	696

input sequence is approximated as $2N \log_2 N$, assuming N is a power of 2. Considering only the remote terminal transmitter, $2N \log_2 N$ instructions are required during every DMT symbol period to modulate the input signal. Given the cyclic prefix length in samples and the system sampling rate, the symbol period can be computed. If the sampling rate of the system is $1/T$ Hz and the number of cyclic prefix samples is ν, then the symbol rate is

$$f_{s,DMT} = \frac{1}{T(N+\nu)}.$$

Therefore, the number of MIPS required to implement the FFT in either the transmitter or receiver is given by

$$N_{MIPS} = \frac{2N \log_2 N}{T(N+\nu)}.$$

Table 3.4 illustrates that, as expected, increasing the FFT size increases the complexity in MIPS of the remote terminal.

3.4 Reference System

Assuming a sub-split HFC configuration as described in Chapter 2, the entire reverse channel spans the frequency band from 5 MHz to 42 MHz. However, designing remote terminals that use the entire 37-MHz band is impractical for a number of reasons. First, as shown by the simulation results in the previous section, the computational complexity of any transceiver operating in 35+ MHz bandwidth can be prohibitive. Based on the complexities detailed in Table 3.4, the only potentially feasible number of subchannels that could be used in a cost-sensitive consumer product is 128 (corresponding to $N = 256$), the computation of which requires nearly 700 MIPS. However, because the required cyclic prefix length is significant in comparison to the symbol period when only 128 subchannels are used, the achievable data rates are significantly lower than when higher numbers of subchannels are used. Hence, to use the limited available reverse channel bandwidth more efficiently, the channel should be partitioned into a larger number of subchannels, which, as Table 3.4 shows, increases the modem complexity and, as a result, the cost. However, aside from the cost issues, it is unlikely that the data rate requirements of the remote terminals warrant use of the entire reverse channel at any time. From Table 1.1, the maximum expected instantaneous bit rate of any service listed is 10 Mbps. If a spectral efficiency of just 2 bits/s/Hz can be achieved in some part of the reverse channel, then a maximum of 5 MHz bandwidth is required to support 10 Mbps (neglecting any excess bandwidth). Hence, none of the remote terminals is ever likely to need access to the entire reverse channel. A more practical solution than designing wide-band transceivers is to segment the reverse channel bandwidth into smaller bands, each of which supports an independent DMT signal.

From the achievable data rates given in Table 3.4, and using the $N = 8192$ case as a reference, it is clear that for any power-to-AWGN ratio the loss in achievable data rate with decreasing FFT size is least severe in terms of both percentage and raw bits per second for $N = 4096$ ($\bar{N} = 2048$ subchannels) and $N = 2048$ ($\bar{N} = 1024$). Hence, assuming that minimizing the cost of the remote terminals is a design goal, dividing the entire reverse channel into 1024 34.5-kHz-wide subchannels appears to be a good compromise between performance and complexity. To ensure the complexity of the remote terminals is not prohibitive, the set of 1024 subchannels can be segmented

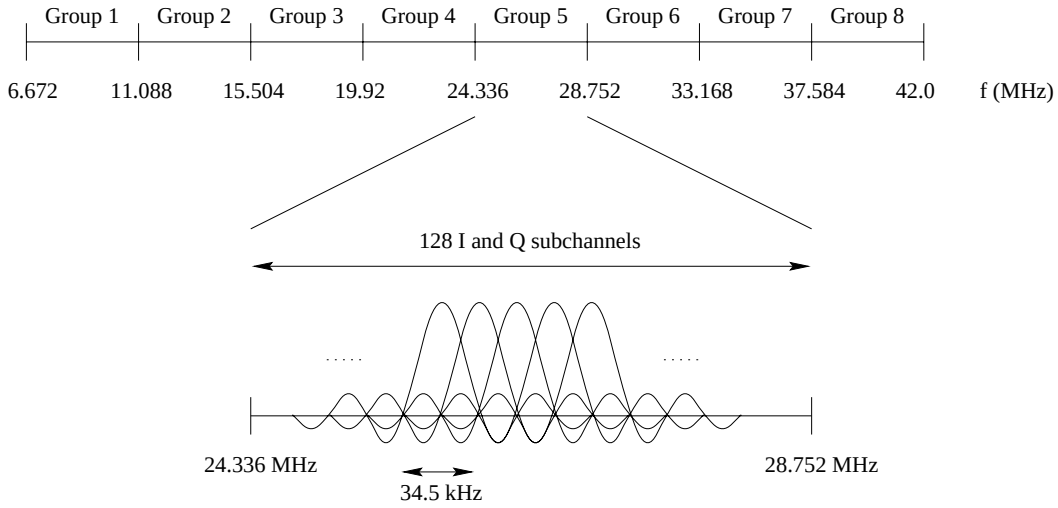

Figure 3.8: Partitioning of reverse channel into eight groups

into eight disjoint *groups* of 128 consecutive subchannels, as shown in Figure 3.8. [22] Given this partitioning, any remote terminal can use a maximum of 128 subchannels that span 4.416 MHz bandwidth at any time. As a result, the maximum sampling rate of the remote terminals must be only 8.832 MHz, and the required FFT size is 256. Each remote terminal configured in this way then has a computational complexity of 131 MIPS, which can be handled easily by a single DSP chip. Furthermore, even less expensive remote terminals can be designed, as will be described in Appendix B. In addition to reducing remote terminal costs, the flexible eight-group configuration offers another advantage. As described in Chapter 2, cable operator studies have revealed that some parts of the reverse-channel bandwidth are poorer quality than other parts. In particular, it is generally agreed that the bandwidth from 5–20 MHz is significantly worse in terms of ingress than is the bandwidth from 15–20 MHz. [12] By designing the remote terminals with frequency-agile carriers, a remote terminal can transmit using the group of subchannels that is best able to support its data rate requirements. Hence, groups that can support high aggregate data rates can be used either by some number of high bit rate users or by a larger number of lower bit rate remote terminals. Those groups that are affected by severe attenuation, passband ripple, or ingress can then be more lightly loaded. Because the average achievable aggregate bit rate in any group is likely to be at least 11 Mbps, as simulation results in

Table 3.5: Reference system parameters

Total number of subchannels	1024
Number of subchannels per group	128
Remote terminal FFT size	256
Headend receiver FFT size	2048
Subchannel width	34.5 kHz
Symbol rate	32 kHz
Symbol period	31.25 μs
Cyclic prefix length	2.26 μs
Cyclic prefix length in samples	20
Remote terminal sampling rate	8.832 MHz
Headend receiver sampling rate	70.656 MHz

the previous section showed, a "typical" 4.416 MHz band in the reverse channel should be able to support any of the services listed in Table 1.1. Furthermore, even though eight independent DMT signals can be transmitted in the eight-group configuration, a single headend receiver, spanning the entire reverse channel bandwidth, can be designed. The receiver performs a 2048-point FFT to demodulate incoming signals, and a controller then ensures signals are decoded correctly. From Table 3.4, 1442 MIPS are required to implement a 2048-point FFT in the receiver. Furthermore, an analog-to-digital converter sampling at over 70 MHz is required, which adds to the receiver implementation costs. However, because the HFC reverse channel is a multipoint-to-point environment, the cost of the headend receiver is amortized over all remote terminals on the network. From this perspective, then, the cost of the broadband receiver is not prohibitive.

Because the eight-group remote terminal configuration offers a number of advantages for HFC reverse channel transmission, it is referred to in the sequel as the *reference configuration*. Table 3.5 summarizes the reference system parameters.

3.5 Summary

The chapter began with a review of Discrete Multi-Tone (DMT) modulation, a transmission technique that partitions a channel into a set of orthogonal subchannels and

allocates data to those subchannels based on their signal-to-noise ratios. The importance of the cyclic prefix and the ease with which it eliminates intersymbol interference was emphasized. The advantages of DMT for transmission on HFC reverse channels were described. In particular, the ability of DMT to transmit only in areas of the spectrum in which meaningful communications can be supported was discussed. Additionally, the adaptivity of DMT was noted as especially advantageous for reverse channel transmission. Equation 3.9 was presented to illustrate the computation of a channel's bit distribution. The equation was subsequently used to compute achievable aggregate data rates under certain transmit power and noise assumptions on the channel modeled in Chapter 2. The simulation results illustrate that DMT can achieve high data rates on HFC reverse channels. Next, the implementational complexity of DMT was evaluated. Based on the evaluation, it was concluded that a remote terminal design spanning the entire reverse channel is impractical. Hence, a reference system was defined, in which the reverse channel is partitioned into a total of 1024 subchannels. The 1024 subchannels are subsequently divided into eight groups of 128 subchannels to enable the design of cost effective remote terminals. The parameters of the reference system were detailed at the end of the chapter for easy reference.

Chapter 4

Synchronized DMT for Multipoint-to-point Communications

The previous chapter described point-to-point DMT, in which a single transmitter communicates with a single receiver over a dedicated channel. In a point-to-point environment, because the DMT transmitter modulates data using an IDFT with a cyclic prefix, the resulting subchannels are independent and orthogonal in the presence of white Gaussian noise. As described in Chapter 2, however, the reverse channel of an HFC network is shared by multiple remote transmitters. Hence, unless precautions are taken, the elegant subchannel orthogonality achieved so easily in point-to-point DMT can be lost in the multipoint-to-point case.

Synchronized DMT (SDMT) is a generalized multipoint-to-point technique introduced by Cioffi in [1] that synchronizes multiple DMT-based remote terminals so they may communicate with a single receiver over a shared channel according to some channel access protocol. SDMT ensures the signal arriving at the common receiver always appears to have been transmitted by a single remote terminal, thus minimizing the increase in receiver complexity required when DMT is used in a multipoint-to-point, rather than point-to-point, environment. A channel access protocol is then used on top of SDMT to coordinate use of the upstream subchannels.

Section 4.1 reviews SDMT. A simple example demonstrates that synchronizing DMT-based remote terminal transmissions preserves the subchannel orthogonality

when DMT is applied in a multipoint-to-point environment. In Section 4.2, under the assumption that synchronized transmissions are required, a procedure for installing DMT-based remote terminals is outlined. Training and retraining procedures, which ensure that the best possible bit distributions are always available to the remote terminals, are presented in Section 4.3.

4.1 Synchronized DMT (SDMT)

To motivate Synchronized DMT (SDMT), consider the following example: Assume that exactly three remote terminals (A, B, and C), located on an HFC network such that their signal propagation times to the headend differ, wish to transmit data to a single headend receiver. To begin, assume that the three remote units have been permanently assigned disjoint sets of subchannels for their upstream transmissions[1], as shown in Figure 4.1. By restricting the remote terminals to transmit on disjoint sets of subchannels, the time-varying nature of the multi-access problem is eliminated for the time being, and synchronization requirements related only to the modulation can be examined. Under this assumption, Figure 4.1 illustrates conceptually the signal that might be received at the headend if Remotes A, B, and C all transmit data for a time equal to one symbol period on their respective subchannel sets beginning at time $t = 0$. Because Remote C is closest to the headend, its signal (information on subchannels marked "C" in the figure) arrives first at the receiver. The signal from Remote B arrives next, and data on subchannels used by Remote A arrives last. Because the remote units "own" their respective subchannel sets, there is no possibility that more than one remote unit transmits at a time on any subchannel. The received time-domain signal is then the sum of the signals transmitted by the individual remote terminals, which by construction contain only frequency components corresponding to the subchannels assigned to each remote unit. As a result, subchannel orthogonality is ensured in this case. Hence, in theory, the remote terminals can transmit as desired on their designated subchannel sets, and the headend receiver can demodulate and reconstruct correctly the original signal provided its sampling rate is

[1]Such a restricted FDMA-like allocation of subchannels for data transmission is undesirable for a number of reasons that will be discussed in the next chapter. However, this configuration helps to demonstrate the need for remote terminal synchronization.

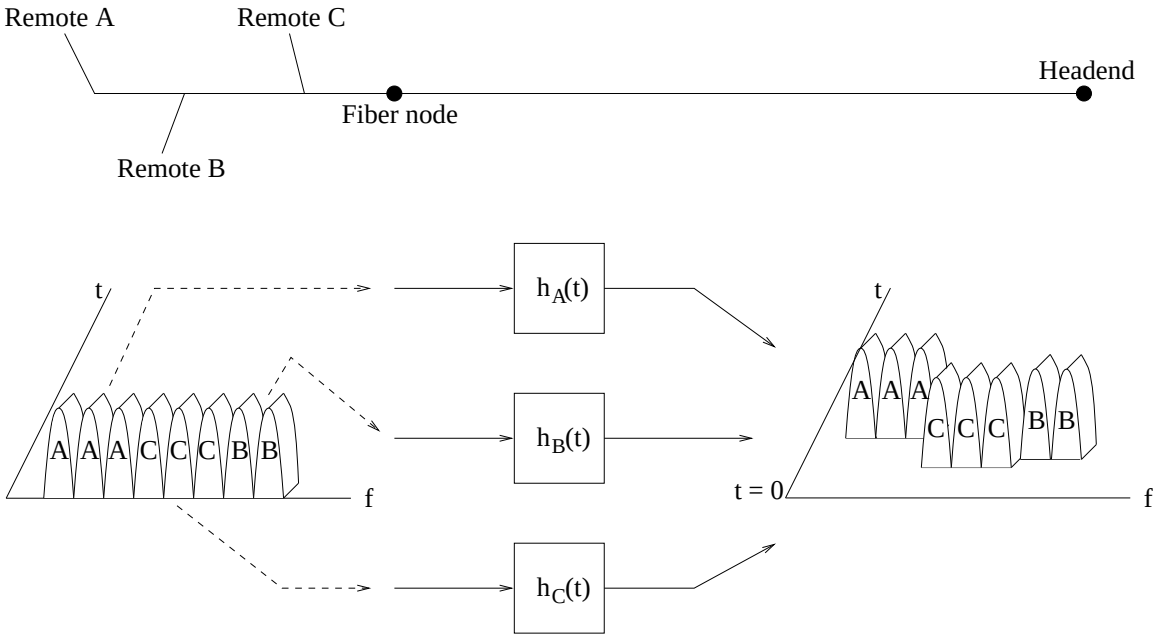

Figure 4.1: Example of multipoint-to-point DMT communications without synchronization

at least twice the bandwidth of the transmitted signals. Thus, when disjoint sets of subchannels are assigned permanently to the remote terminals, synchronization of the remote terminals is not required; however, it dramatically simplifies the complexity of the headend receiver. From the discussion in Chapter 3, the DMT receiver is a block processor: it collects a number of time-domain signal samples and manipulates these samples to reconstruct the set of QAM subsymbols that was transmitted during a symbol period. Hence, the DMT receiver must define where the collection of each block of samples begins and ends. Ideally, the receiver begins collecting data samples immediately after the cyclic prefix samples, which are, in general, corrupted by interference from the preceding symbol, as described in Chapter 3. In a multipoint-to-point configuration in which multiple remote terminals transmit on permanently-assigned disjoint subchannel sets, however, if transmissions from the remote units are not synchronized to arrive at the headend at the appropriate times, then the cyclic prefix samples corresponding to the various subchannel sets most likely do not coincide in time in the received signal; hence, "the cyclic prefix" no longer exists as a distinct, well-defined set of received samples that can be discarded in the receiver. Instead,

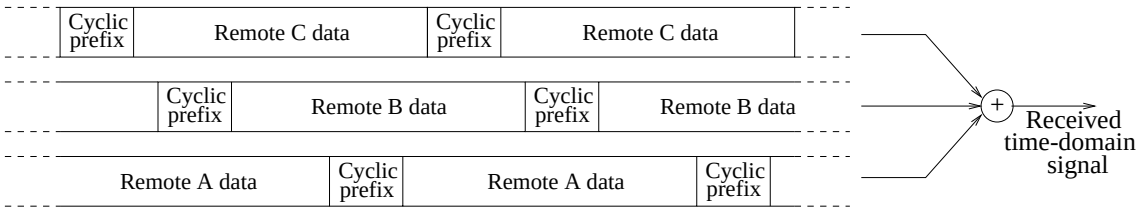

Figure 4.2: Loss of well-defined cyclic prefix when unsynchronized remote terminal transmissions are allowed in multipoint-to-point configurations

cyclic prefix samples transmitted by the various remote terminals may be scattered throughout the received data. Figure 4.2 illustrates the scattering of the cyclic prefix assuming the three-remote configuration shown in Figure 4.1. Because the received signal is the sum of unsynchronized remote terminal transmissions, the individual cyclic prefixes cannot be removed from the received signal. Hence, use of a cyclic prefix does not help to eliminate ISI if the remote terminal transmissions are not synchronized. Instead, to mitigate ISI a bank of parallel equalizers (one equalizer per subchannel) must be implemented in the receiver, which dramatically increases the complexity of the headend modem. Moreover, as mentioned in Chapter 3, equalization is a suboptimal detection technique, so the performance of an unsynchronized equalized system may be degraded with respect to a synchronized system that preserves the cyclic prefix.

The preceding discussion presumed that disjoint subchannel sets are permanently assigned to the remote terminals, implying that no additional channel access protocol is required: remote terminals simply transmit as desired on their respective subchannel sets, and no bandwidth conflicts occur. However, some networks might require a less restrictive allocation of the available bandwidth, particularly if remote terminal transmission requirements vary significantly with time. Hence, a more sophisticated channel access protocol may be necessary to coordinate use of the subchannels. If so, then synchronization of remote terminal transmissions becomes even more critical. Because the remote terminal distances from the headend differ, ensuring for all time that no more than one remote terminal uses any particular subchannel is extremely difficult without synchronization. In addition, the cyclic prefix problem described previously also exists when a less restrictive allocation of subchannels is allowed with

unsynchronized transmissions.

If remote terminal transmissions are synchronized so that the symbol boundaries of signals arriving at the headend receiver coincide, however, then the key properties of point-to-point DMT also hold for multipoint-to-point transmission. This observation led Cioffi to propose Synchronized DMT (SDMT). [1] The general idea of SDMT is to coordinate remote terminal transmissions so that the signal arriving at a single receiver appears to have been transmitted by a single remote terminal. The receiver then proceeds, for the most part, as if it were in a point-to-point environment. To ensure that symbols transmitted by the various remote terminals on an HFC network arrive at the headend with synchronized symbol boundaries, remote units must delay their transmissions a number of samples inversely proportional to their physical distances (in terms of length of coaxial cable) from the fiber node. As a result of the synchronization, the cyclic prefix as defined for point-to-point transmission is preserved; hence, ISI can be eliminated just as in the point-to-point case by discarding the cyclic prefix in the receiver, and no equalizer other than the FEQ is required. Figure 4.3 illustrates this concept for the three-remote configuration considered previously. Furthermore, if more than one remote terminal transmits during a particular symbol period using disjoint sets of subchannels, then the headend receiver can decode the received signal by applying the appropriate mixture of FEQ taps, assuming the headend controller knows which remote transmitted on each subchannel. Finally, the overhead required to initiate and maintain synchronized remote terminal transmissions is reasonable, as will be shown subsequently.

Synchronization of remote terminal transmissions to enable SDMT can be achieved by a two-step procedure. Initially, remote terminals must synchronize their local clocks to a master clock so that all remote units transmit with respect to a common reference. The master clock must be broadcast by the headend controller in a downstream control channel: because of the limitations imposed by the HFC configuration (see Chapter 2), broadcasting the clock in a downstream channel is the only means by which all remote terminals can access a common clock signal. In addition, each remote terminal must acquire a sample delay, computed by the headend controller when the terminal is installed on the network, before it can begin operating. A remote terminal's required sample delay is the difference between the maximum round-trip signal delay to and from the headend of all remote units on the network and the

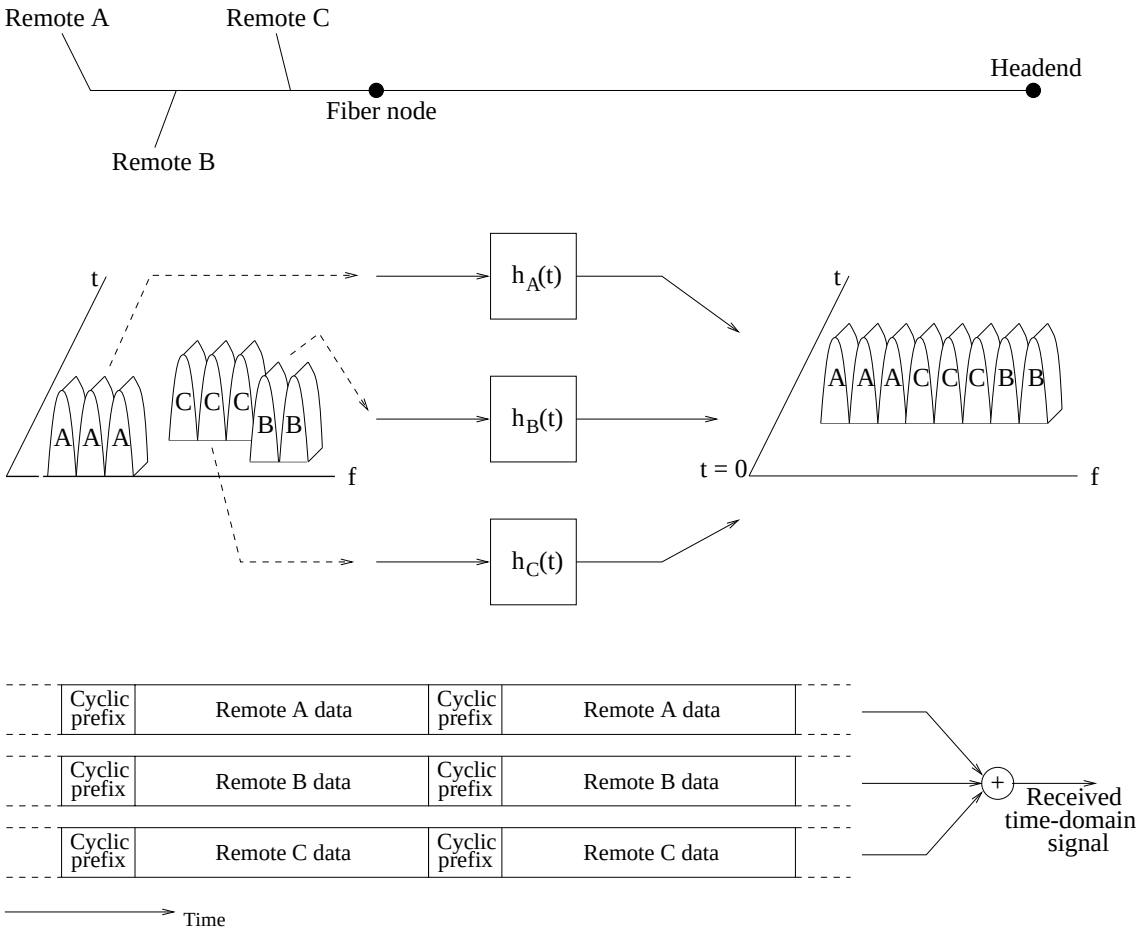

Figure 4.3: Preservation of the cyclic prefix in multipoint-to-point environments by delaying remote terminal transmissions

remote terminal's round-trip signal delay.

4.2 Remote Terminal Installations

Installation of a remote terminal on a multiaccess network is the process of physically connecting the terminal to the network and following some procedure to condition it for coordinated multipoint-to-point upstream communications. One installation task resulting from the requirements outlined in Chapter 1 is the exchange of a terminal's factory-assigned serial number for a shorter network node address, which enables

more efficient communications from the headend to the remote terminal. Also, as discussed in the preceding section, SDMT-based remote units must be synchronized during installation.

When a remote unit is first connected to the network, its installation request must not interfere with ongoing data signals from remote terminals already installed and synchronized. The new remote can receive and decode information in the downstream control channel, but, because it does not yet have a node address on the network and the headend controller does not know its factory-assigned serial number, the remote terminal cannot respond to a poll from the headend. Furthermore, because the terminal does not know its distance from the headend modem, it cannot be synchronized precisely without ranging, which requires the transmission of a probing signal by the remote. Hence, there are two practical installation alternatives: transmitting installation signals in a dedicated band, or using the same bandwidth that is used for data transmissions. Under normal circumstances, remote unit installations are infrequent events. For this reason, and because the upstream bandwidth is a limited resource, dedicating some portion of the reverse channel bandwidth to installations is undesirable: the reserved bandwidth would be idle most of the time. Furthermore, given the anticipated ingress noise problems described in Chapter 2, locating a portion of the reverse channel bandwidth that is always able to support communications would be difficult. Installing remote terminals using the same bandwidth that supports data transmissions is a more robust solution, and it is also a more efficient solution if the percentage of symbols allocated for installations can be changed according to demand. However, to avoid interfering with the remote units currently operating, newly connected units cannot transmit installation requests unless the other remotes are silent. To accommodate installation requests using the same bandwidth that is used for data transmissions, silent periods must be provided periodically in the upstream data path. The duration of these silent periods must be sufficient to ensure that an installation signal transmitted by any remote on the network arrives at the headend modem before the signal corresponding to the next non-installation symbol arrives. Specifically, the silent interval must be at least the maximum variation in round-trip signal delay over all remotes plus the time required to transmit the installation information. Because the maximum variation in round-trip delay must be known to determine the required length of the installation interval, it is necessary to

know the maximum signal delay when configuring the network. Note that because the fiber trunk is common to all remote terminals, the maximum signal delay is incurred by the remote terminal furthest, in terms of length of coaxial cable, from the fiber node. Hence, given a particular installation interval duration, a maximum network span is imposed.

4.2.1 Synchronization Procedure

To begin its synchronization procedure, a newly-connected remote terminal first loop-times its local clock with the master clock signal broadcast by the headend transmitter. It then monitors the downstream control channel to learn its group assignment and to detect the trigger indicating the presence of an installation period, which tells installing units to transmit their installation parameters. In general, the group assignment will be the group least loaded by remote terminals currently transmitting data. Technically, the only required installation parameter is the remote unit's factory-assigned serial number to enable the headend to communicate with it, although other parameters could also be transmitted. The serial number is exchanged for a shorter node address, which is used subsequently by the headend modem to send addressed messages directly to the remote in the downstream control channel. Because the bit capacities of the subchannels in the assigned group from the installing remote to the headend have not yet been determined, the remote unit uses four-level differential QPSK (2 bits per subchannel) with 100% redundancy to transmit its installation signal on some subset (or perhaps all) of the available subchannels for two symbols during the next installation period. Assuming remote terminal serial numbers are 48 bits long, then at least 48 subchannels must be used to send installation signals with this scheme. When the headend controller detects this signal, it compares the incoming signal's symbol boundaries, which are determined by when the receiver detects energy during the installation period, to a reference signal. In general, there will be a difference in symbol boundaries because the remote terminal is not yet synchronized, and, using the downstream control channel, the controller computes and sends the delay in samples required of the synchronizing unit to correct the symbol boundary misalignment. The remote unit then implements the computed sample delay and transmits during the next installation period a signal requesting verification that it

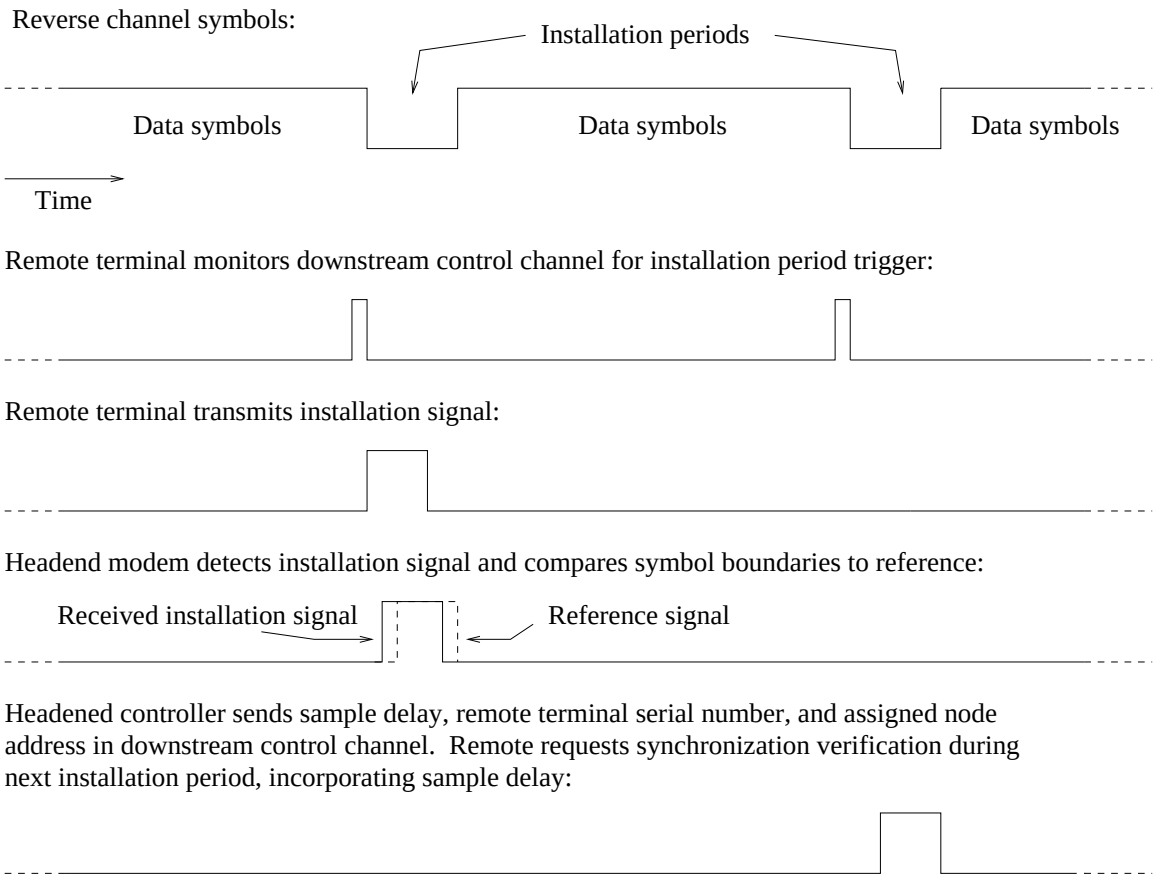

Figure 4.4: Synchronization procedure for SDMT remote terminals

is synchronized. If the remote unit transmission is indeed synchronized, the headend controller sends a signal to that unit in the downstream channel to indicate that no further shifting is required, and that the remote unit may now communicate with the headend modem incorporating the appropriate delay. Otherwise, the synchronization procedure is repeated. Figure 4.4 illustrates the synchronization process after a remote terminal has loop-timed its local clock to the master clock.

After the initial sample delay has been determined, unless a remote unit is moved or its connection to the network is terminated, it should not have to be resynchronized. Failing to synchronize the remote units to within a certain tolerance can result in interchannel interference, which can decrease the achievable bit rates on some subchannels. The effects of inaccurate synchronization and ways to relax the

synchronization requirements are addressed in Appendix B.

Using the installation technique described, there is a nonzero probability that multiple remote units attempt installation during the same installation period. Without loss of generality, assume exactly two remote terminals transmit installation signals during the same installation interval. One of two situations then occurs. On some networks, it may be possible for both installation requests to arrive intact at the headend if the difference in propagation times from the remote terminals to the headend is greater than the time required to transmit an installation signal. In this case the headend controller may either install one of the remote terminals and ignore the other, or, if the headend modem has queueing capabilities, it may be able to queue the second request until it has completed installation of the first remote. Although this scenario is theoretically possible, on practical HFC networks it is not likely to occur. As described in Chapter 2, the fiber trunk, which is up to 50 miles long, is common to all remote terminal-to-headend transmission paths. The only difference in propagation times is introduced by coaxial branches, which span up to 2 miles. Hence, if a propagation time through coaxial cable of 7.5 μs/mile is assumed, the maximum difference in propagation times is 15 μs. Because transmission of installation signals during two consecutive symbols is required for the headend receiver to decode the transmissions, the maximum DMT symbol period that would enable two remote terminals to transmit successful installation signals during the same installation period is 3.75 μs, which corresponds to a symbol rate (and subchannel width) of nearly 270 kHz. Given the characteristics of the HFC reverse channel, use of such a high symbol rate is unlikely to provide subchannels with flat or nearly flat SNRs across their bandwidths. The reference system, for example, uses a symbol rate of 32 kHz, which corresponds to a symbol period of 31.25 μs. In this case, if multiple installation signals are transmitted during a single installation period, then all transmissions interfere with each other. The headend receiver detects a collision when it is unable to decode the received signal, and the headend controller then sends a collision message in the downstream control channel to inform all remotes that attempted installation during the previous installation period that their attempts were unsuccessful. The remote terminals then wait a random number of installation periods before trying again.

Installations are infrequent events: each remote terminal must be installed only

once, assuming it is not disconnected from the network. Hence, the probability of collision during any installation period can be made arbitrarily small by increasing the frequency of the installation periods. Of course, the penalty for increasing the frequency of installation periods is a decrease in the time the channel is available for data transmissions. Because installation times of less than a few seconds should be acceptable to subscribers, a small to moderate collision probability can probably be tolerated. As an example, the reference system symbol rate imposes a minimum installation period length requirement of three symbols (two symbols for transmission of the installation signal, plus one symbol to allow for the maximum difference in reverse channel propagation times). If an overhead of 0.1% of the total number of symbols is allowed for installations, then installation periods occur every 2997 symbols, or 10.7 times per second. This frequency of installation periods should enable the remote terminals to synchronize in less than one second.

4.3 Training and Retraining

After a new remote terminal has been installed, its transmissions are synchronized with transmissions from other remote terminals. Hence, with permission from the headend controller it can now use symbols in the upstream data path for upstream communications without interfering with ongoing data transmissions. However, the subchannel capacities and FEQ tap settings for the new remote terminal need to be determined before the remote can most efficiently use whatever upstream bandwidth is allocated to it in the future. Thus, after completing the installation process, each remote terminal must undergo a training procedure.

After verifying that the new remote terminal is properly synchronized, the headend controller sends an addressed message in the downstream control channel instructing the new remote unit to transmit a "wide-band"[2] signal during a specified number

[2]The term "wide-band" here means that the remote unit uses all allowed subchannels in its bandwidth during a specified number of upcoming training periods. Note that some subchannels may be disallowed for some reason. For example, if the network operator has another system in place on the network using bandwidth that overlaps the bandwidth of the DMT system, then the overlapping DMT subchannels are marked by the headend controller as disallowed. This information is then sent to the remote terminals in the downstream control channel, and the affected subchannels are never used for either training or data transmission.

of upcoming training periods to train the headend receiver. To ensure accurate bit capacities and FEQ taps are computed by the headend modem, several consecutive training intervals are used for the initial training of newly installed remote terminals. Installed remote units remain quiet while the new remote unit transmits a training signal on the permissible subset of subchannels in its group, and the headend controller records the bit capacity and FEQ tap setting of each subchannel from that remote unit. The bit capacities are used to determine subchannel assignments when the remote terminal later requests data transmissions.

As the network operates, both the reverse channel frequency response and noise profile can change. As described in Chapter 3, in point-to-point DMT these changes are accommodated by a bit swap algorithm, which tracks symbol-to-symbol changes in subchannel SNRs and redistributes some or all bits carried by a degrading subchannel to one or more other subchannels that can better support those bits. In a multipoint-to-point environment, however, use of a bit swap algorithm may not be sufficient to compensate for changes in the channel and noise. Because the channel is shared by multiple remote terminals with varying data rate requirements, if a flexible channel access protocol is used, then at least some low-rate remote terminals will be assigned bandwidth for short periods of time or very infrequently. In such cases, a bit swap algorithm may be too slow to modify the bit distributions adequately. Furthermore, running bit swap algorithms to adjust the bit distributions of all remote terminals on the network is burdensome to the headend controller. Consequently, to accommodate changes in the channel and noise characteristics, remote terminals are retrained periodically during another interval reserved specifically for this purpose. As during the installation and training periods, all remote terminals that are not retraining remain quiet to allow the headend controller to update the bit capacities and FEQ tap settings for the retraining remote. Typically, the bit distribution is updated by estimating the SNR of each subchannel and noting any differences between the new bit distribution and the current one. An adaptation algorithm such as LMS can be used to update the FEQ taps. Depending on the frequency of the retraining intervals, the number of remotes on a particular network, and other system parameters, each remote can be retrained as often as many times per second or as infrequently as every two (or more) seconds. As an example, in the reference system, assuming training intervals are 20 symbols long and 0.25% of the total symbols are

reserved for training, then training intervals occur every 7980 symbols, or 4 times per second. If up to 64 remote terminals are assigned to each group, then using only 0.25% of the total symbols for retraining implies that each remote modem can be retrained every 16 seconds. If 1% of the total symbols are reserved for retraining, then retraining periods occur every 1980 symbols, or about 16 times per second. In this case, again assuming up to 64 remote terminals are assigned to each group, every remote terminal can be retrained every 4 seconds.

4.4 Summary

The chapter began with a review of Synchronized DMT, in which multiple DMT-based remote terminals are synchronized to enable them to communicate with a common receiver over a shared channel according to some channel access protocol. An example was given to illustrate that synchronizing DMT-based remote terminal transmissions preserves the desired subchannel orthogonality when DMT is applied in a multipoint-to-point environment. Hence, it was concluded that synchronizing the remote terminals is beneficial from the perspective of system complexity. Next, under the assumption that SDMT is desired, a procedure for installing DMT-based remote terminals was outlined. New procedures for training and retraining remote terminals were presented to enable efficient communications according to a channel access protocol.

Chapter 5

Design and Analysis of SDMT-based Random Access Protocols

After a remote unit has been installed and the optimal bit distribution has been determined, the remote terminal is ready to request upstream communications according to a channel access protocol. This chapter examines several protocol alternatives. Because the reverse channel bandwidth is limited, and the instantaneous data transmission requirements of the remote terminals may vary significantly from a few hundred bits per second to several megabits per second, the reverse channel bandwidth must likely be allocated dynamically to the remote units.

In Section 5.1, several generalized channel access protocol alternatives are reviewed. First, Time-Division Multiple Access (TDMA), Frequency-Division Multiple Access (FDMA), Aloha, Carrier-Sense Multiple Access (CSMA), and Busy-Tone Multiple Access (BTMA) protocols are described and evaluated as candidates for controlling HFC reverse channel access. It is concluded that none of these alternatives is a satisfactory solution for HFC reverse channels. Reservation-based protocols, however, offer significant advantages for HFC reverse channel transmission, including accommodating the required overhead periods for installation, training, and retraining. Consequently, new reservation-based protocols designed specifically for multicarrier remote terminals are presented. Section 5.2 describes one such protocol, called the Reservation-Based Multicarrier protocol with TDMA data symbol

Figure 5.1: Time-division multiple access

assignments (RBM-T). The performance of the RBM-T protocol, in terms of how well it enables the channel to be used and the delays messages incur due to the protocol, is evaluated via analysis and simulation. The results suggest that a variation on the RBM-T protocol might offer improved performance. Consequently, in Section 5.3 the Reservation-Based Multicarrier protocol with combined TDMA and FDMA data symbol assignments (RBM-TF) is described. Simulation results are presented to illustrate that the RBM-TF protocol outperforms the RBM-T protocol.

5.1 Generalized Protocol Alternatives

Protocols can be divided into two classes: those with centralized control and those with distributed control. With centralized control, a single node coordinates the activities of all other nodes on the network. In protocols with distributed control, all nodes behave according to the same set of rules. [23]

Perhaps the simplest distributed-control protocol alternatives for the reverse channel of HFC networks are *Time-Division Multiple Access (TDMA)* and *Frequency-Division Multiple Access (FDMA)*, both of which ensure that transmissions from remote terminals do not interfere with each other. Hence, TDMA and FDMA are known as *conflict-free* protocols. In TDMA, one or more consecutive symbol periods are grouped together into *slots*. These slots are then assigned in a round-robin fashion to the remote terminals on the network, and the probability of conflicts in channel use is theoretically zero. Assuming there are M remote terminals on a network, each terminal is assigned every Mth slot, as shown in Figure 5.1. If a remote terminal does not require the use of a slot assigned to it, then the channel is idle during that slot. Conversely, if a remote terminal cannot fit its data transmission into one slot, it must wait $M-1$ additional slot periods for its next assigned slot before it can send the remainder of its message. To implement TDMA in a network, each remote terminal only needs to maintain a slot counter that ranges from 1 to M to determine when it

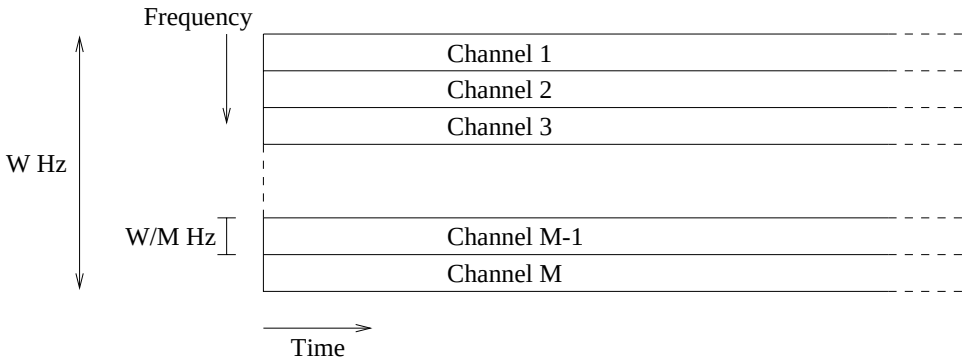

Figure 5.2: Frequency-division multiple access

may transmit. Thus, TDMA is simple and inexpensive to implement.

In FDMA, the channel bandwidth is divided into a set of parallel, equal-bandwidth, slower subchannels, each of which supports an identical modulation. Each subchannel is assigned permanently to a distinct remote terminal on the network, which ensures that no data transmission conflicts occur. Again assuming the network serves M remote units, the channel is divided into M subchannels, as illustrated in Figure 5.2. Each remote terminal has immediate access to its assigned subchannel. If a remote terminal does not need to transmit any data, then its subchannel remains idle. One advantage of FDMA relative to TDMA is that the transmit bandwidth of each remote terminal is reduced to only W/M Hz, where W is the total available bandwidth, including excess bandwidth. Hence, the required sampling rate of the remote terminal analog-to-digital converters is only $2W/M$ Hz as opposed to $2W$ Hz. The disadvantage of FDMA with respect to TDMA is that the average time required to transmit messages is generally longer. The reason is that a remote terminal with data ready to transmit on a TDMA network must wait an average of $M/2$ slot periods before the arrival of its designated slot, at which time it can transmit its message. However, when the terminal's next slot period arrives, the message is transmitted as quickly as possible, using the entire available W Hz. Thus, assuming each message requires one TDMA slot for transmission, the average time to transmit a message is $M/2 + 1$ slots. On an FDMA network, a remote terminal can begin to transmit a message immediately, but the time required to transmit that message is M times longer than in the TDMA case because only $1/M$th of the total bandwidth is used. Consequently,

assuming again that each message requires one TDMA slot for transmission, the average message transmission time on an FDMA network with the same total bandwidth is equal to M TDMA slots. Only if $M = 2$ are the average times required to complete transmission of a message in the TDMA and FDMA cases the same. For any $M > 2$, the average transmission time is less for TDMA.

Although both TDMA and FDMA are simple to implement and ensure that all transmissions are free from interference from other transmissions, they are inflexible. Assuming the channel can support a total of R bits/second, the maximum data rate that can be transmitted by each remote terminal is R/M bits/second for either TDMA or FDMA. As discussed in Chapter 1, some services HFC networks may need to support are bursty in nature, suggesting that both the bandwidth requirements of individual remote terminals and the aggregate traffic load may vary dramatically as a network operates. Hence, a static allocation of the available bandwidth is undesirable. Instead, remote terminals should be able to secure bandwidth proportional to their needs.

An alternative channel access protocol that meets the requirement of dynamic bandwidth allocation is *Aloha*, which was developed by Abramson in the late 1960s at the University of Hawaii. [24] In contrast to TDMA and FDMA, Aloha does not guarantee that transmissions from independent remote terminals do not interfere with each other. On the contrary, remote units on an Aloha network simply transmit messages as necessary, without regard to the requirements or activities of the other remote terminals. Hence, Aloha enables dynamic bandwidth allocation at the expense of conflict-free data transmission. A remote terminal that has transmitted a message waits to receive an acknowledgement from the receiver that its message arrived intact. If two or more terminals transmit at the same time, then a *collision* occurs, and the remote terminals involved in the collision retransmit their messages after a randomly-chosen delay. If no collision occurs, then the receiver sends an acknowledgement message back to the transmitter to indicate the message was received successfully. It can be shown easily that if message transmissions (both new messages and those that have been rescheduled due to collisions) are assumed to be attempted by an infinite population of remote terminals at an aggregate rate of G messages per unit time, where G is Poisson distributed, then the channel can be used for successful transmissions a maximum of only 18% of the time. By slotting the time axis and

restricting the remote units to transmit synchronized to the slot boundaries, which results in the protocol known as *slotted Aloha*, Roberts showed that the maximum fraction of time the channel is used for successful data transmissions (often called the *channel utilization* or *throughput*) increases to 36%. [25] Other variations of Aloha have been developed, but none enables a throughput greater than 36%.

One disadvantage of Aloha protocols when considering the reverse channel of HFC networks is the low achievable channel utilization. Although the reverse channel may span up to 37 MHz, a maximum throughput of 36% means that effectively only 13.3 MHz is used at most. Hence, nearly 2/3 of the available bandwidth is wasted under the best of circumstances. A second problem with Aloha for HFC reverse channels is caused by the electrical length of typical HFC networks. Assuming reasonable propagation times for coaxial cable and fiber of 7.5 μs/mile and 5.5 μs/mile, respectively, the round-trip propagation time on an HFC network can approach 0.5 milliseconds. Consequently, remote terminals using Aloha for upstream HFC transmission must wait up to 0.5 ms to learn the outcomes of their transmissions. If remote terminals are required to verify that each message they transmit has arrived successfully before they may transmit additional messages, then a lag time between successive message transmissions of up to 0.5 ms is imposed automatically, thus decreasing the maximum rate at which remote units may transmit data. If a message transmission is unsuccessful, then the corresponding remote terminal incurs at least the round-trip delay plus an additional delay due to rescheduling the transmission. Furthermore, a message could collide multiple times before being received successfully. Hence, the average message transmission time when an Aloha protocol is used is likely to be excessive. Finally, pure Aloha and slotted Aloha protocols are notoriously unstable. However, they can be stabilized in theory, although not necessarily in practice, by imposing on the remote terminals a message retransmission rule that takes into account the past and present channel activity.

Based on the preceding discussion of Aloha protocols and the potential increases in message transmission delays that can result due to collisions between messages, a conflict-free protocol is desirable for upstream HFC transmission. One family of protocols that attempts to avoid message collisions by forcing remote terminals to consider the state of the channel (in use or idle) before transmitting messages is the class of *Carrier-Sense Multiple Access (CSMA)* protocols. CSMA protocols are

distributed-control protocols that were developed by Tobagi and Kleinrock to offer channel utilization values higher than those achievable using Aloha protocols. [26] Under CSMA protocols, a remote terminal desiring to transmit a message first checks the channel state to determine whether another remote terminal is already transmitting. If the channel is sensed busy, then the remote unit reschedules its transmission for a later time according to a retransmission rule. At that time it again checks the channel state and proceeds accordingly. If the channel is sensed idle, then the remote terminal transmits its message. Because a finite time, due to the propagation time of the transmission medium, is required for the other remote terminals on the network to detect this transmission and thereby recognize that the channel is busy, there is a nonzero probability that another remote terminal will begin to transmit a message that interferes with the first message. If a collision occurs, retransmission of the messages involved is required. However, because the remote terminals transmit messages only if they sense the channel is idle, the channel utilization increases significantly with respect to Aloha, where users transmit without regard to the channel state. The achievable channel utilization of CSMA is a function of the electrical length of the network and the size of transmitted messages, but for networks with low round-trip propagation delays with respect to the message transmission time, the channel utilization can exceed 80%. Furthermore, by slotting the time axis and forcing the remote terminals to transmit synchronized to the slot boundaries, the channel utilization can be improved even further.

Although CSMA protocols can provide high channel utilization, use of CSMA on the HFC reverse channel is not possible. As described in Chapter 2, HFC hardware prevents the remote terminals from sensing the state of the reverse channel. However, a variation of CSMA called Busy-Tone Multiple Access (BTMA), proposed by Tobagi and Kleinrock, incorporates the "listen before transmitting" principle for networks on which remote terminals cannot directly sense the state of the shared channel. [27] In BTMA, a separate overhead channel that can be received by all remote terminals is used to indicate the state of the shared channel. Remote terminals desiring to transmit data then monitor the overhead channel to determine whether or not to transmit their messages. Signals on the overhead channel can be generated in a variety of ways, depending on the characteristics of the network. For example, a remote terminal that is transmitting a message could simultaneously transmit a busy

tone signal in the overhead channel. Alternatively, the recipient of the message being transmitted could generate the busy tone. This technique would be necessary for HFC networks because remote terminals can only transmit in the reverse channel bandwidth and receive in the downstream bandwidth. Hence, the headend modem would need to transmit the busy tone signal. However, the headend modem cannot generate the busy tone until it has received a signal in the upstream bandwidth, by which time the transmission for which the busy tone should be generated is long over because the electrical length of the network is so large. The only way BTMA could be used to coordinate HFC reverse channel transmissions is if a channel sensing device is installed at the fiber node. To minimize maintenance costs, installing nonessential equipment at the fiber node is undesirable. Hence, BTMA is not an attractive option for controlling remote terminal transmissions in the reverse channel.

CSMA and BTMA protocols force remote terminals to transmit data while attempting to avoid collisions. However, collisions between messages can and do occur. *Reservation-based protocols*, in contrast, guarantee conflict-free data transmissions by requiring all remote units to reserve transmission bandwidth via a centralized scheduling mechanism. Hence, reservation-based protocols provide conflict-free data transmissions by imposing centralized control. A remote terminal needing to transmit a message first transmits a reservation request to the central controller. The reservation request may be sent in a separate channel, or it could be transmitted using the same channel that is used for data transmissions. Furthermore, depending on the implementation of the protocol, the reservation requests themselves could be subject to collisions, or they may be conflict-free. Upon receiving a reservation request, which must somehow identify the requesting remote terminal and convey its transmission requirements, the centralized controller then schedules future bandwidth for use by the requesting remote terminal and transmits the assignment to the terminal. Because a central controller schedules all data transmissions, collisions are avoided. The penalty for ensuring collision-free data transmissions, however, is a minimum delay of one round-trip propagation time before a remote terminal can begin to transmit its message.

Traditional reservation-based protocols may feature reservation slots followed by data transmission slots, or a portion of the total bandwidth may be reserved for reservation request transmissions. The latter approach is not practical for HFC reverse

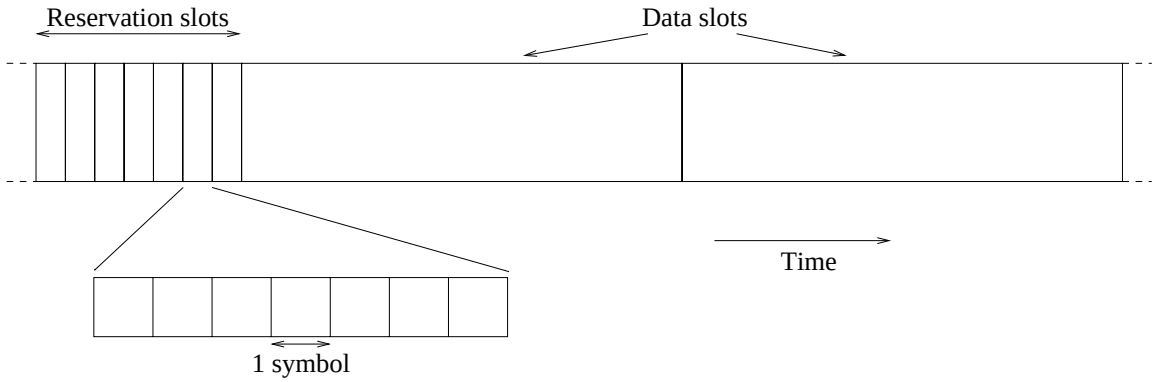

Figure 5.3: Illustration of reservation and data slots with single-carrier modulation

channels because ingress or channel nulls can render large portions of the spectrum useless. The problem is compounded by the unpredictability of poor-quality regions in the reverse channel. Hence, as with remote terminal installations, the same bandwidth that is used for data transmissions is also used to transmit reservation requests in this design. In single carrier systems, several symbols are grouped into a reservation slot, as shown in Figure 5.3. As shown in Chapter 3, however, the number of bits supported by a single symbol in most DMT systems is far greater than the number required to request a reservation, especially if the channel is partitioned into a large number of subchannels. The same is generally true for other multicarrier modulations, too. This observation motivates the design of reservation-based channel access protocols specifically for multicarrier-based remote units. In this dissertation, the general class of reservation-based protocols for multicarrier-based remote units will be called *Reservation-Based Multicarrier (RBM)* Protocols.

Because the reverse channel traffic load may vary significantly as the network operates, of concern is what percentage of the available channel bandwidth should be allocated to reservation slots as opposed to data transmission slots. Ideally, the number of reservation slots should change to best accommodate the needs of the remote terminals. Perhaps the simplest approach is to allow the headend controller to label each symbol as either reserved or available for contention. Contention symbols can be used by remote terminals to transmit reservations to request bandwidth for data transmissions. By transmitting the symbol status information to the remote

units in a downstream control channel, the headend controller informs the remote units of the channel use. Because reservation requests are transmitted only during contention symbols, they do not collide with data transmissions, assuming information transmitted in the downstream control channel is received and decoded perfectly. The implications of errors in decoding the downstream control channel are addressed in Appendix B.

5.2 The RBM-T Protocol: A Reservation-Based Multicarrier Protocol with TDMA Data Symbol Assignments

Figure 5.4 illustrates a simple protocol based on the idea that the headend controller schedules all data transmissions by communicating to the remote terminals when they may transmit what type of information. As described briefly at the end of the last section, each symbol in the upstream data path is classified as either a contention (that is, unassigned for data transmissions) symbol or as some other type of symbol, and this information is transmitted to the remote units by the headend controller via a downstream control channel. In general, most symbols that have not been assigned for data transmissions are classified as contention symbols, although, as described in Chapter 4, some symbols must be reserved for installation, training, and retraining remote terminals. During contention symbols, the remote terminals are authorized to transmit reservation requests, which are, in this protocol, always unsolicited. A reservation request contains the requesting terminal's node address and a description of the unit's transmission needs. Because the headend controller has designated which symbols are contention symbols, it identifies signals received during contention symbols as reservation requests and proceeds accordingly. After a reservation request is received successfully by the headend modem, upcoming symbols are assigned to the remote terminal that transmitted the successful reservation. The assignment is transmitted to the remote terminal in the downstream control channel. To minimize bookkeeping for the headend controller, symbols assigned for data transmission are assigned to a single remote unit. Hence, no two remote terminals can transmit data during the same symbol, and therefore a TDMA-like data symbol assignment

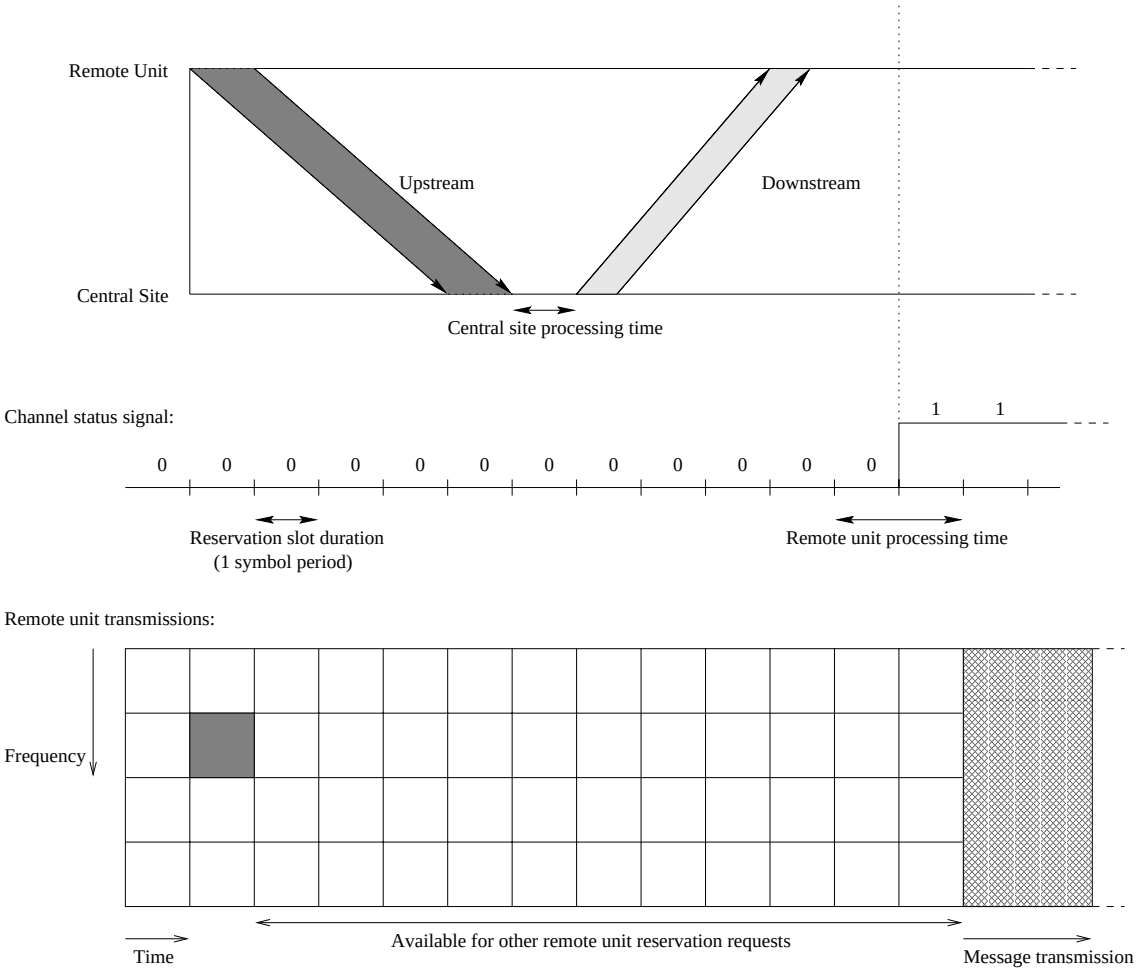

Figure 5.4: Illustration of RBM-T protocol with $K = 4$

is used. The TDMA-like data symbol assignments enable the headend controller to inform the remote terminals of the status (contention or reserved) of upcoming symbols by broadcasting a channel status signal, which ensures none of the remote terminals transmits a reservation request during a symbol that has been reserved by the headend controller for data transmission, installations, or training. Because data symbols are assigned entirely to only one remote terminal, this protocol is called the *Reservation-Based Multicarrier protocol with TDMA data symbol assignments*. The name is abbreviated to *RBM-T* for simplicity.

Technically, a transmitter partitioning a channel into as few as two subchannels is

considered multicarrier. Hence, using DMT as an example, an FFT size of 4 produces two subchannels, each of which supports a number of bits proportional to its SNR. Using a larger FFT size increases the number of subchannels into which the channel is partitioned and, in general, the number of bits supported by each DMT symbol. As the FFT size increases, however, so does the symbol period. Consequently, systems using large FFT sizes support large numbers of bits per symbol, but their symbol rates are low. Based on this observation, multicarrier symbols may support many more bits than necessary for the transmission of a reservation request, particularly if the channel is partitioned into a large number of subchannels. Consequently, in the RBM-T protocol each contention symbol is split into a set of K concurrent but disjoint reservation slots, as shown in Figure 5.4. A reservation slot is a group of subchannels that together support the number of bits required to transmit a reservation. Hence, if a channel supports different numbers of bits per subchannel, then the K groups may be composed of different numbers of subchannels. However the subchannels are grouped into reservation slots, the partitioning must be observed by all remote units to ensure that collisions overlap completely. To mitigate any interchannel interference problems due to slight offsets in remote terminal carrier frequencies, the subchannel sets should be composed of adjacent subchannels. Depending on the number of subchannels into which a channel is partitioned, the bit capacities of those subchannels, and the number of bits required to represent a reservation request, the value of K could be as small as 1. If $K = 1$, each symbol, even though it is a multicarrier symbol, can support only one reservation request. For this reason, and because symbols are assigned in a TDMA manner for data transmissions, the $K = 1$ case can be considered a *slotted single carrier* version of the RBM-T protocol. It will illustrate the advantages or disadvantages of multicarrier modulation with respect to single-carrier modulation under the RBM-T protocol.

During a contention symbol, a remote unit that needs to request a reservation chooses randomly from the set of K reservation slots and transmits its request on that subchannel set. If no other remote unit transmits an overlapping reservation request, then the headend receiver successfully decodes the request, and the headend controller assigns one or more upcoming symbols to the remote unit for transmission of the data corresponding to the reservation. The particular remote unit to which

the data symbols have been assigned learns of the assignment when it receives an addressed packet in the downstream control channel indicating the indices of upcoming symbols that have been assigned to it. For any value of K, if two or more reservation requests collide on, for example, the ith subchannel set, then the headend controller broadcasts in the downstream control channel a message indicating that whichever remote terminals transmitted requests on the ith subchannel set during that symbol period must retransmit their requests. The remote terminals involved in the collision then reschedule their requests for a later time according to a rescheduling algorithm, at which time the procedure of checking the channel state and either rescheduling or randomly choosing a subchannel set and transmitting the reservation request is repeated.

To enable the headend modem to decode reservation requests transmitted during contention symbols, two requirements must be satisfied. First, the bit distributions used by the remote terminals must be the same *when reservation requests are transmitted*. Hence, the "lowest common denominator" bit distribution is used for reservation requests. Without this restriction, the constellation used on a particular subchannel could change depending on which remote terminal transmitted a reservation request. Because reservation requests are unsolicited, if the bit distributions were not the same, the headend receiver would be unable to determine how many bits were transmitted on each subchannel. Similarly, the second requirement is also a consequence of the unsolicited requests. If a single symbol period is used to transmit reservation requests, which is desirable to minimize the probability of collisions between requests and thus maximize the efficiency, all remote terminals must apply a coarse "inverse FEQ" to their reservation requests. Because the reverse channels from the various remote terminals to the headend may differ significantly, the gains and phase shifts introduced by the subchannels can also vary from remote terminal to remote terminal. Again, the headend receiver is unable to tell which remote terminal transmitted a reservation request, and it thus cannot apply the correct mixture of FEQ taps to the received signal in the decoding process. If a coarse inverse FEQ is implemented in the remote terminals, however, and the number of bits transmitted on each subchannel is set to some value lower than could be supported (that is, a noise margin is imposed), then decoding the received reservation requests should be straightforward. As an example, based on the results in Chapter 3, which showed

that most regions of the worst-case reverse channel can support at least 3 bits, transmitting reservation requests using 2 bits per subchannel and applying a coarse inverse FEQ in the remote terminals should result in essentially error-free decoding.

The inverse FEQ is computed from the updated receiver FEQ taps after a remote terminal's training interval has ended. Because the inverse FEQ is used only when reservation requests are transmitted, the taps are computed under the assumption that each remote terminal transmits only 2 bits per subchannel; hence, the required precision is not as high as for the receiver FEQ. The tap settings are then transmitted to the appropriate remote terminal in the downstream control channel.

If implementation of inverse FEQs is not feasible for a system, then remote terminals must transmit their reservation requests using differential QPSK (DQPSK) to enable the headend receiver to decode the signals. Use of DQPSK decreases the efficiency of the protocol because successful transmission of a reservation request requires two consecutive symbol periods during which no interfering request is transmitted. First, depending on the channel state, two consecutive symbol periods may not be available, which imposes a delay while the remote terminal waits for two consecutive contention symbols in the upstream data path. Furthermore, if two consecutive contention symbols are available but a collision occurs during either of the two symbol periods, then the reservation must be retransmitted. Because application of an inverse FEQ results in a more efficient protocol, it is the method assumed in the remainder of this dissertation. The impact of the inverse FEQ on the system complexity is evaluated in Appendix B.

5.2.1 Protocol Evaluation Criteria

As mentioned in Section 5.1, of interest for evaluating the performance of a particular protocol is the expected channel utilization or *throughput* it supports at various traffic loads. Again, the throughput is defined as the percent of time the channel is used for successful data transmissions. Figure 5.5 illustrates the channel activity as a function of time for the RBM-T protocol when the sum of the round-trip propagation delay plus the headend unit and remote unit processing times is equal to the time required to transmit a message. To simplify notation, all time variables are normalized by the message transmission time. Hence, the message transmission time is set equal to 1,

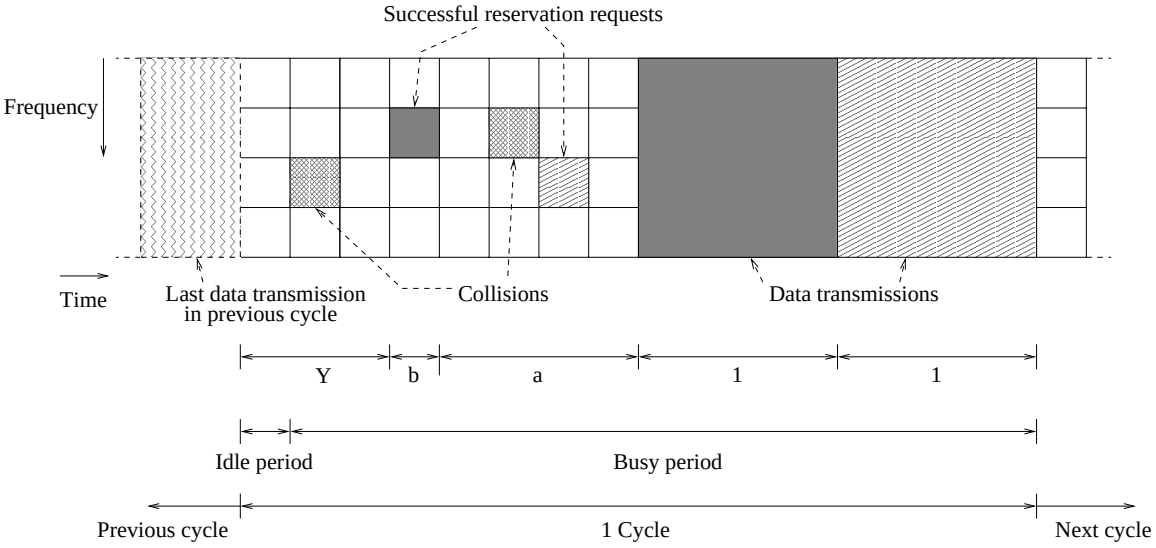

Figure 5.5: Example of potential channel activity under the RBM-T protocol with $a = 1$

and the sum of the round-trip propagation time plus the remote unit and headend modem processing times normalized by the message transmission time is denoted by a.

Referring to Figure 5.5, a *cycle* consists of at least one successful reservation request and its corresponding data transmission plus the *idle period* between the last data transmission in the previous cycle and the first reservation attempt (successful or not) in the current cycle. A *busy period* is defined as the time from the first reservation attempt after the idle period to the end of the last data transmission in the current cycle. When $a \leq 1$, the "last data transmission" is the data transmission corresponding to the last successful reservation request occurring during the period a.[1] Thus, every busy period is composed of an equal number of successful reservation requests and data transmissions, and each busy period ends with a data transmission. Using these definitions, and denoting the average durations of a busy period and an

[1]When $a > 1$, the RBM-T protocol analysis becomes much more difficult. As a result, only the $a \leq 1$ case is considered in this dissertation.

idle period as $\bar{B}$ and $\bar{I}$, respectively, the throughput is

$$S = \frac{\bar{U}}{\bar{B} + \bar{I}}, \tag{5.1}$$

where $\bar{U}$ is the average time per cycle the channel is used for data transmissions. Hence, the throughput is just the average fraction of each cycle that is used for data transmission.

The *channel access delay* is defined as the average time from when a remote unit first has a message ready for transmission to when transmission of that message begins. The access delay may be caused by a number of factors. First, under any reservation-based protocol, all messages incur a minimum delay that is the sum of the time required to transmit each reservation request, the times required for the headend to process each reservation request and for the remote unit to process the corresponding grant message, and the round-trip propagation time of the network itself. A potential second cause of delay is the state of the channel when a remote unit attempts to schedule a reservation request. For example, in the RBM-T protocol, if the channel is in use for data transmission (that is, it is busy) when the remote wishes to transmit a reservation request, then the remote unit must reschedule its request for a later time, thus incurring a delay. Alternatively, if the channel is idle but two or more remote units transmit reservation requests that overlap in time and frequency, then the requests collide, and the remote units involved in the collision must reschedule their requests for a later time.[2] The *transport delay* of a particular message is the access delay plus the time required to transmit the message. If all messages require the same amount of time for transmission, then the transport delay is simply the access delay plus a constant. In general, for protocols allowing conflicts between either data messages or reservation requests, as the throughput increases, the expected access and transport delays also increase.

[2]It is worth noting here that the delay incurred because of colliding reservation requests can be significantly larger than the delay when the channel is found busy because the remote units must wait the round-trip propagation time before they can determine their requests collided. Only then can they reschedule their requests for a later time according to the delay distribution. For networks that span large distances, as most HFC networks do, it is especially desirable to avoid reservation request collisions and the corresponding increases in access delay.

5.2.2 Throughput Analysis

To simplify the RBM-T throughput analysis, the transmission channel from each remote terminal to the headend is assumed to be flat and identical, and all remote units are assumed to be the same distance (in terms of transmission line length) from the headend. Hence, the bit tables of all remote units are the same: all subchannels support B bits, and each symbol, regardless of which remote unit transmitted it, is composed of $\bar{N}B$ bits. Furthermore, although the network traffic will in general consist of both constant bit rate and packetized data, for simplicity the analysis assumes only packetized data is supported. It is also assumed that all data messages are the same length and that all reservation requests are the same length. The message length is chosen to require an integral number of multicarrier symbols for transmission so that no bandwidth is wasted by constraining the multicarrier system to use all subchannels in a symbol for data transmission as required by the RBM-T protocol. Clearly, if this requirement is not satisfied, then the protocol could be very inefficient, since it is possible that only a small percentage of a symbol might be needed for data transmission. Finally, the analysis assumes that all remote units satisfy the requirements for SDMT: they are loop-timed to a master clock broadcast by the headend transmitter and impose the appropriate sample delays to ensure their transmissions arrive at the headend synchronized to predefined symbol boundaries. A method for achieving SDMT was described in the Chapter 4.

Under the RMB-T protocol, the aggregate network traffic is composed of both packetized data transmissions and requests for bandwidth. Constant-length messages are assumed to be generated by an infinite population of remote units at a collective rate of λ messages per second according to some probability distribution. The assumption of an infinite population is equivalent to assuming that each remote terminal attempts to transmit only one message. In other words, the delay between remote terminal message requests is infinite. Denoting the message transmission time by T, messages and their corresponding reservation requests are generated by the population of remote units at a rate of $S = \lambda T$ per message transmission time, which is the throughput of the network because a message is either transmitted successfully or rescheduled, but never discarded. Because some of the transmitted reservation requests will collide and require retransmission, reservation requests, both new and retransmitted, and their corresponding messages are *offered* at an average aggregate

Table 5.1: Notation

a	=	Round-trip propagation delay plus headend and remote unit processing times, normalized to message transmission time
k	=	Number of subchannels required to send reservation request
K	=	Number of request slots per multicarrier symbol
b	=	One symbol period, normalized to message transmission time
$\bar{N}$	=	Number of subchannels in multicarrier system
G	=	Average number of reservation requests scheduled per message transmission time, including rescheduled requests
S	=	Channel throughput (Alternatively, the rate at which new messages are generated)
L	=	Number of multicarrier symbols required to transmit each message
N_s	=	Number of additional successful reservation requests during first-success symbol period
N_a	=	Number of successful reservations during a
N_r	=	Number of reservation requests attempted during a symbol period
$\bar{Y}$	=	Expected time from the end of the last message in the previous cycle to the first successful reservation request in the current cycle

rate of G per time T, where $G > S$. For the purpose of analysis, G is assumed to be Poisson distributed.

For reference, the notation used in the analysis is detailed in Table 5.1.

From the definition of a cycle, it is clear that there must be at least one successful reservation request and its corresponding data transmission in every cycle. After the initial successful reservation in a cycle, while the first remote unit waits for a message in the downstream control channel granting use of the channel for data transmission, other remotes may transmit reservation requests during the following a time units. Furthermore, if $K > 1$ it is possible for more than one successful reservation request to occur during the symbol period in which the "first" success occurs. Thus, to compute the expected time the channel is used for data transmission, it is necessary to know the expected number of additional successful reservation requests (N_s) occurring during the *first-success* symbol period and the expected number of successful reservations

during a (N_a). Then, in normalized time units, $\bar{U}$ is simply

$$\bar{U} = 1 + E\{N_s\} + E\{N_a\}. \tag{5.2}$$

$E\{N_a\}$ can be computed by considering what happens during a single multicarrier symbol period. The number of reservation requests attempted during a symbol period is denoted as N_r, which is a Poisson distributed random variable in accordance with the traffic assumptions. If $K = 1$, which means that each contention symbol is composed of only one request slot, then the expected number of successes during a symbol period is $\Pr\{N_r = 1\}$. If $K > 1$, a Bernoulli random variable X_i, $i = 1 \ldots K$ is defined as

$$X_i = \begin{cases} 1 & \text{if exactly one reservation request is transmitted on subchannel set } i \\ 0 & \text{otherwise} \end{cases}$$

Because the remote terminals transmitting reservation requests choose randomly from the K request slots, it is sufficient to consider only the ith slot. Then the expected number of successes during a contention symbol period is $KE\{X_i\}$. Because X_i is a Bernoulli random variable, its expected value is $\Pr\{X_i = 1\}$, which can be computed by conditioning on the total number of reservation requests attempted during a symbol period:

$$E\{X_i\} = \Pr\{X_i = 1\} = \sum_{n=1}^{\infty} \Pr\{X_i = 1 | N_r = n\} \Pr\{N_r = n\}.$$

Because the offered traffic is Poisson distributed with rate G reservation requests per message transmission time,

$$\Pr\{N_r = n\} = \frac{e^{-bG}(bG)^n}{n!},$$

where b denotes the symbol period in normalized time units. $\Pr\{X_i = 1 | N_r = n\}$ is found by observing that $X_i = 1$ when n reservation requests are attempted during that symbol period if and only if exactly one remote unit transmits its reservation request on subchannel set i and the other $n-1$ remote units transmit their reservation requests on any of the remaining $K-1$ subchannel sets. Appendix C shows that there

are $\begin{pmatrix} n+K-1 \\ n \end{pmatrix}$ ways to distribute n requests among K subchannel sets. Hence, there are $\begin{pmatrix} n+K-3 \\ n-1 \end{pmatrix}$ ways to distribute $n-1$ requests among $K-1$ subchannel sets, and

$$\Pr\{X_i = 1 | N_r = n\} = \frac{\begin{pmatrix} n+K-3 \\ n-1 \end{pmatrix}}{\begin{pmatrix} n+K-1 \\ n \end{pmatrix}}.$$

Thus,

$$E\{X_i\} = \Pr\{X_i = 1\} = \sum_{n=1}^{\infty} \frac{n(K-1)}{(n+K-1)(n+K-2)} \cdot \Pr\{N_r = n\},$$

and, during any contention symbol,

$$E\{N_{\text{successes}}\} = KE\{X_i\} = \begin{cases} \Pr\{N_r = 1\} & \text{if } K = 1 \\ K(K-1)\sum_{n=1}^{\infty} \frac{n}{(n+K-1)(n+K-2)} \cdot \Pr\{N_r = n\} & \text{if } K > 1 \end{cases}$$

Therefore, if each message requires L symbols for transmission, then a is aL symbols in duration, and

$$E\{N_a\} = \begin{cases} aL \Pr\{N_r = 1\} & \text{if } K = 1 \\ aLK(K-1)\sum_{n=1}^{\infty} \frac{n}{(n+K-1)(n+K-2)} \cdot \Pr\{N_r = n\} & \text{if } K > 1 \end{cases} \tag{5.3}$$

$E\{N_s\}$ can be computed in a similar manner. In this case, however, it is known that, because this symbol period is the first-success period, at least one successful reservation request occurs, which leaves $K-1$ concurrent request slots in which additional successful reservation requests can be transmitted. If $K = 1$, then clearly $E\{N_s\} = 0$. If $K = 2$, then the expected number of additional successes during this symbol period is 1 if exactly 2 reservation requests are attempted; otherwise it is zero. Hence, the expected number of additional successes is just $\Pr\{N_r = 2\}$ if $K = 2$. To determine the expected number of additional successes during the first-success symbol

period for values of K greater than 2, a Bernoulli random variable W_i, $i = 1 \ldots K-1$, is defined:

$$W_i = \begin{cases} 1 & \text{if exactly one reservation request is transmitted on subchannel set } i \\ 0 & \text{otherwise} \end{cases}$$

Then, similar to before,

$$E\{W_i\} = \Pr\{W_i = 1\} = \sum_{n=1}^{\infty} \Pr\{W_i = 1 | N_r = n\} \Pr\{N_r = n\},$$

but this time

$$E\{N_s\} = (K-1)E\{W_i\}.$$

Clearly, if the total number of attempts N_r during this symbol period is 1, then the expected number of *additional* successes is zero. If $N_r = 2$, then both reservation requests must be successful because it is known that at least one success occurs during this symbol period. Thus, $\Pr\{W_i = 1 | N_r = 2\} = \frac{1}{K-1}$. For $N_r \geq 3$, the probability that exactly one of the remaining $(n-1)$ reservation requests is transmitted on the ith subchannel set is

$$\Pr\{W_i = 1 | N_r = n, n \geq 3\} = \frac{\begin{pmatrix} n+K-5 \\ n-2 \end{pmatrix}}{\begin{pmatrix} n+K-3 \\ n-1 \end{pmatrix}}$$

by the same arguments that were used previously. Hence, for $K \geq 3$, the expected number of additional successes can be written as

$$E\{N_s\} = \Pr\{N_r = 2\} + (K-1) \sum_{n=3}^{\infty} \frac{\begin{pmatrix} n+K-5 \\ n-2 \end{pmatrix}}{\begin{pmatrix} n+K-3 \\ n-1 \end{pmatrix}} \cdot \Pr\{N_r = n\}.$$

Finally, simplifying and collecting the expressions for the expected number of additional successes during the first-success symbol period for various values of K yields

$E\{N_s\}$:

$$E\{N_s\} = \begin{cases} 0 & \text{if } K = 1 \\ \Pr\{N_r = 2\} + & \\ \quad (K-1)(K-2)\sum_{n=3}^{\infty} \frac{n-1}{(n+K-3)(n+K-4)} \cdot \Pr\{N_r = n\} & \text{if } K \geq 2 \end{cases} \tag{5.4}$$

Substituting the appropriate versions of Equations 5.3 and 5.4 into Equation 5.2 yields $\bar{U}$ for any value of K.

Next, the expected duration of a cycle is computed as

$$\bar{B} + \bar{I} = 1 + a + b + E\{N_a\} + E\{N_s\} + \bar{Y}. \tag{5.5}$$

Both $E\{N_a\}$ and $E\{N_s\}$ were found in the derivation of $\bar{U}$. $\bar{Y}$ is the expected time from the end of the last message transmission in the previous cycle to the first successful reservation request(s) in the current cycle. Hence, $\bar{Y}$ includes both the idle period and the expected time from the first reservation request attempt in the cycle to the first *successful* reservation request transmission, and it can be computed by considering each symbol period to be an independent Bernoulli trial. The assumption of independence is valid when an infinite population of remote units is assumed: because the rescheduling of reservation requests is infinite, the activity during any symbol is independent of the outcomes of preceding symbols. Returning to the Bernoulli trials, the outcome of each symbol period is a success if at least one reservation request is transmitted successfully during that symbol period; otherwise, it is a failure.[3] Let A be the event that at least one successful reservation request occurs during a symbol period. The sequence of symbol outcomes is a geometric random variable, and the expected number of symbols before at least one successful reservation request occurs is $\frac{1-\Pr\{A\}}{\Pr\{A\}}$. Normalized by the number of symbols per message,

$$\bar{Y} = \frac{1 - \Pr\{A\}}{L \Pr\{A\}}. \tag{5.6}$$

[3]Note that in this case a symbol is termed a failure if there are no reservation request attempts or if all attempts during a symbol period fail.

$\Pr\{A\}$ can be computed by conditioning on the number of reservation requests attempted during a symbol period. Given that n reservations are attempted, the probability that at least one success occurs is simply the probability that at least one of the subchannel sets is chosen by exactly one remote unit. If $K = 1$, then $\Pr\{A\}$ is $\Pr\{N_r = 1\}$ because only one success per symbol period can occur. For $K > 1$, define A_i as the event that subchannel set i is chosen by exactly one remote unit. Then $\Pr\{A|N_r = n\}$ is just $\Pr\{A_1 \cup A_2 \cup \cdots \cup A_K|N_r = n\}$. Invoking inclusion-exclusion,

$$\Pr\{A_1 \cup A_2 \cup \cdots \cup A_K|N_r = n\} =$$
$$\sum_i \Pr\{A_i|N_r = n\} - \sum_{i<j} \Pr\{A_i \cap A_j|N_r = n\} + \sum_{i<j<l} \Pr\{A_i \cap A_j \cap A_l|N_r = n\} - \ldots$$

The closed-form solution is

$$\Pr\{A_1 \cup A_2 \cup \cdots \cup A_K|N_r = n\} = \sum_{i=1}^{\min(n,K)} (-1)^{i+1} \frac{\left(\begin{array}{c} K \\ i \end{array}\right)\left(\begin{array}{c} n+K-(2i+1) \\ n-i \end{array}\right)}{\left(\begin{array}{c} n+K-1 \\ n \end{array}\right)},$$

where $\left(\begin{array}{c} c \\ 0 \end{array}\right)$ is taken to be 1, regardless of the value of c.[4] Removing the condition on the number of reservation requests attempted yields

$$\Pr\{A_1 \cup A_2 \cup \cdots \cup A_K\} = \Pr\{A\} =$$
$$\sum_{n=1}^{\infty} \left[\sum_{i=1}^{\min(n,K)} (-1)^{i+1} \frac{\left(\begin{array}{c} K \\ i \end{array}\right)\left(\begin{array}{c} n+K-(2i+1) \\ n-i \end{array}\right)}{\left(\begin{array}{c} n+K-1 \\ n \end{array}\right)} \cdot \Pr\{N_r = n\} \right]. \quad (5.7)$$

Equation 5.7 can be evaluated for any number $K > 1$ of subchannel sets (request

[4]Note that when $n = K$, the second combination in the numerator evaluates to $\left(\begin{array}{c} -1 \\ 0 \end{array}\right)$ when $i = \min(n, K)$, which makes no sense. However, writing a separate formula valid for $n = K$ is cumbersome and can be avoided if $\left(\begin{array}{c} -1 \\ 0 \end{array}\right)$ is taken to be 1.

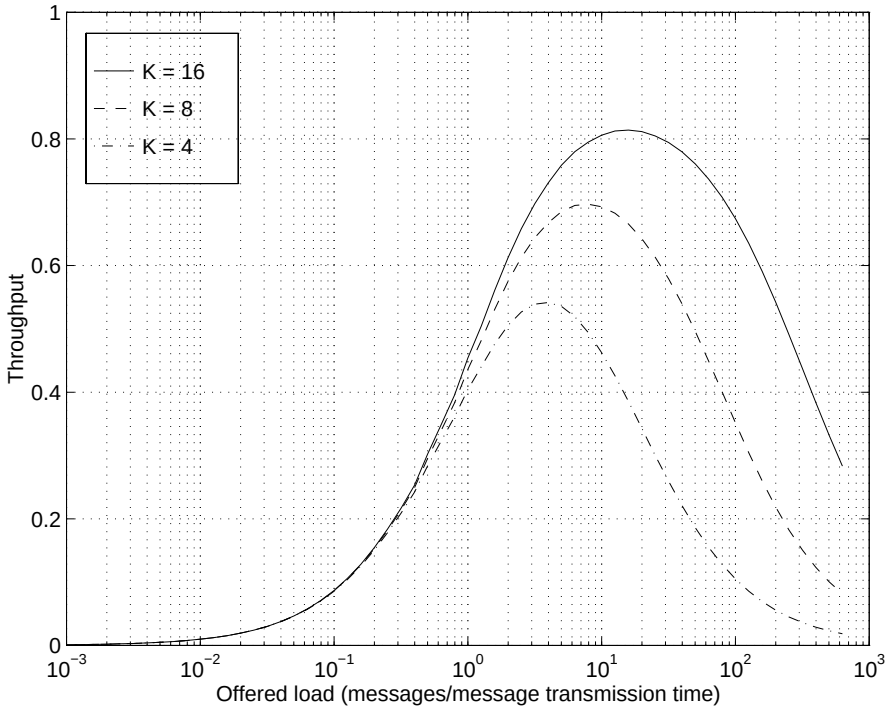

Figure 5.6: Throughput as a function of offered load for various reservation request sizes

slots). As found previously, for $K = 1$, $\Pr\{A\} = \Pr\{N_r = 1\}$. The appropriate value of $\Pr\{A\}$ is then substituted into Equation 5.6 to find $\bar{Y}$. By substituting the results of Equations 5.3, 5.4, and 5.6 into Equation 5.5, the expected duration of a cycle can be computed. Finally, the throughput is found from Equation 5.1.

Figure 5.6 illustrates the throughput as a function of the offered load G for various reservation request sizes and a fixed message size. The total number of subchannels was set to 64, and the number of request slots per symbol was set to 4, 8, and 16. The figure shows that as the size of the reservation request messages increases (that is, K decreases), the maximum throughput decreases because more collisions occur among requests.

Figure 5.7 shows the effect of the number of subchannels on the achievable throughput. This time the message and reservation request sizes were fixed, and the number of subchannels per symbol and, as a result, the symbol period were changed. The number of subchannels was varied from 16 to 64, and the number of subchannels per

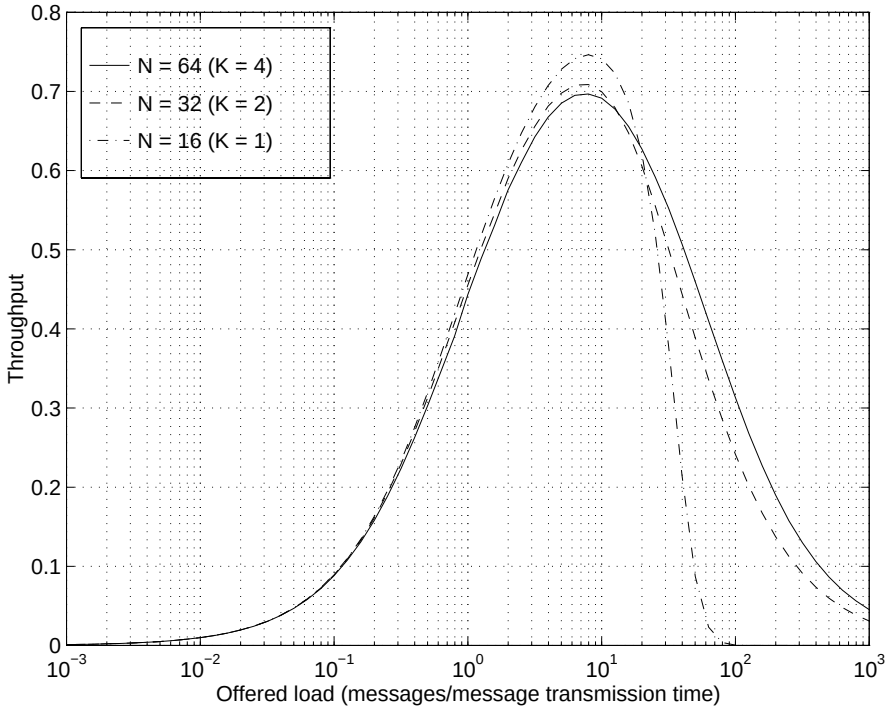

Figure 5.7: Throughput as a function of offered load parameterized by the number of subchannels available

reservation request was fixed at 16. (Hence, K varied from 1 to 4.) The number of symbols per message was set to 2 for $K = 4$. The figure shows that at offered loads less than some critical value, the $K = 1$ curve stays higher than the other two, indicating that the slotted single-carrier version achieves a higher throughput than the higher-order configurations. Beyond the critical load value, however, the multicarrier configurations perform better.

Insight into the differences between the $K = 1$ (slotted single-carrier) and $K > 1$ (multicarrier) versions of the RBM-T protocol can be fostered by considering the channel activity at various points on the curves in Figure 5.7. As a simple example, an offered load of 1 message per message transmission time means that during any time period equal to the message transmission time, an average of 1 remote unit becomes ready to transmit a reservation request. As described previously, ready remotes check the channel state and either transmit or reschedule their reservation requests based on the channel state. According to Figure 5.7, when the offered load is 1 message per

message transmission time, the throughput achieved by any version of the protocol is just under 0.5, which means that if 2 message "slots" are monitored, on average 1 of the 2 will carry a message. The other message slot is available for reservation request transmissions by ready remote units, and, to maintain a throughput of about 0.5, an average of 1 successful reservation request occurs during this message slot. Because the offered load is only 1 message per message transmission time, which means that on average only one remote terminal attempts to transmit a reservation request during each message slot, collisions between reservation attempts are rare for any value of K, and it follows that approximately 1 successful reservation request will occur during each available message transmission slot.

Considering now an offered load of 10 messages per message transmission time, the throughputs for the various versions of the protocol vary between roughly 0.69 ($K = 4$) and 0.74 ($K = 1$). The reason for the differences in throughput is illustrated in Figure 5.8, which shows the expected number of successful reservation requests during a message transmission time. From Figure 5.8, at an offered load of 10 the expected number of successful reservation requests per message slot when $K = 1$ (slotted single-carrier) is greater than the expected number of successful reservation requests per message slot when $K > 1$. The reason is that the probability of more than one reservation attempt during a request slot (that is, the probability of collision) at this offered load when $K = 1$ is only about 0.12. Consequently, every attempted reservation request is much more likely to succeed than to collide, because each is probably the only reservation request transmitted during that symbol. In contrast, when $K = 4$ the probability of 2 or more reservation attempts during a symbol period (which consists of 4 request slots) is approximately 0.71. Even if exactly 2 reservation requests are attempted during a symbol period, the probability that both succeed is only 0.75. Thus, whereas in the $K = 1$ case a reservation request has a success probability of almost 0.9 at this offered load, when $K = 4$ the probability of success drops significantly, as does the expected number of successful reservation requests during a message slot. The consequence of the decrease in expected number of successes during a is a corresponding decrease in the throughput. Note that at an offered load of approximately 20 the curves in both Figures 5.7 and 5.8 cross, and at loads above this point the expected number of successes per message slot and throughput are greatest for $K = 4$. Thus, at high offered loads, when the probability

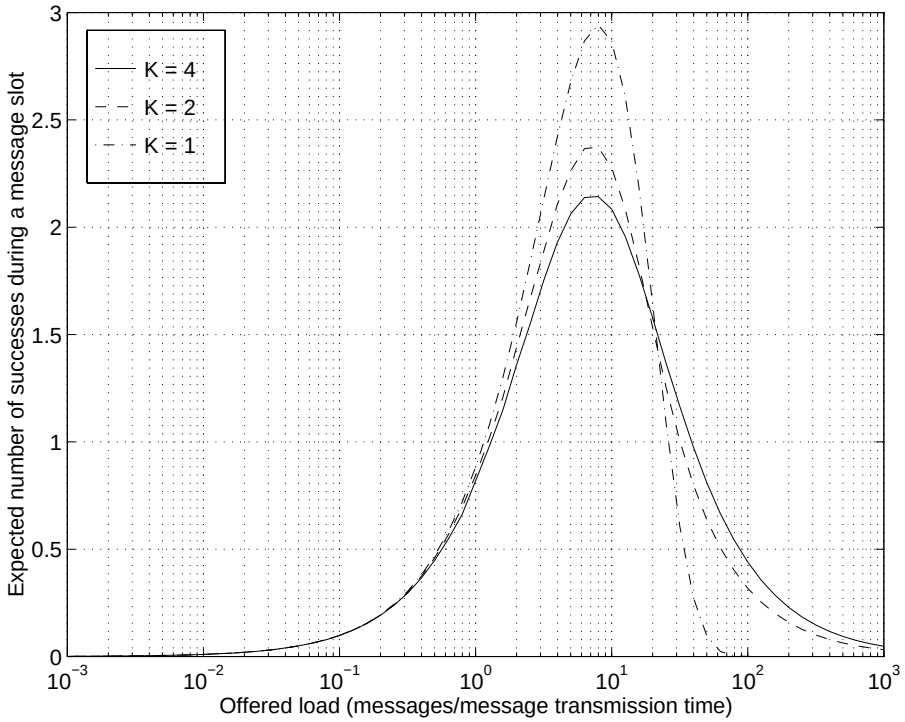

Figure 5.8: Expected number of successes per message slot for 32-subchannel ($K = 4$), 16-subchannel ($K = 2$), and slotted single-carrier ($K = 1$) channel access protocols

of request collisions in the $K = 1$ case exceeds the probability of collision in the $K > 1$ cases, the availability of multiple request slots during each symbol period dramatically increases the expected number of successful reservation requests that occur and, therefore, the throughput.

5.2.3 Simulation Results: Throughput-Delay Curves

The analysis in the previous section illustrates the throughput that can be achieved by various versions (that is, different values of K) of the RBM-T protocol subject to the following set of assumptions:

- The population of remote terminals is infinite, and rescheduling delays are infinite;
- All transmissions consist of constant-length messages;

- No constant bit rate traffic is transmitted;
- The offered traffic is Poisson-distributed;
- The sum of the round-trip propagation time and the transceiver processing times is less than or equal to the message transmission time.

In reality, none of these assumptions is likely to hold. First, the assumption of an infinite population of remote terminals with infinite rescheduling delays forces the symbol outcomes to be independent. As a result, an analysis of the maximum achievable throughput was possible. In a real HFC network, of course, message delays can be critically important, and one goal of the system design is to ensure a tolerable message delay while achieving high throughput. Second, a flexible HFC network should allow transmission of messages of various sizes and should also support constant bit rate services. In addition, real-world offered traffic is probably not well-approximated by a Poisson distribution, especially because traffic sources of terminals using the reverse channel could be anything from a video game (short, high-priority, bursty messages) to an Ethernet node (longer, bursty messages with varying priorities) to a constant bit rate connection (constant-length, high-priority, regular messages). Finally, given the physical size of HFC networks, it is unlikely that the message transmission time will always exceed the sum of the round-trip delay plus the transceiver processing times. As an example, assume the reference system is deployed on an HFC network that imposes a 0.5 ms round-trip delay, including transceiver processing times.[5] If each of the 128 subchannels in a group can support only two bits, then each DMT symbol from a remote terminal contains 256 bits. Because the symbol rate is 32 kHz, the round-trip delay corresponds to 16 DMT symbols, which implies that meeting the $a \leq 1$ requirement imposes a minimum message size of 16 symbols, or at least 4096 bits, if the most efficient use of the channel is desired. Restricting messages to such a large minimum length is likely to result in network inefficiencies; some upstream messages require many fewer than 4096 bits. Clearly, increasing the symbol period (that is, using narrower subchannels) increases the minimum message size necessary to meet the $a \leq 1$ constraint. In addition, channels that can support higher-order constellations on the subchannels also increase the minimum required message size. For

[5]For ease of terminology, the sum of the round-trip propagation time plus the remote terminal and headend modem processing times is referred to henceforth as the round-trip delay.

example, Chapter 3 showed that HFC reverse channels should be able to support well over 2 bits per subchannel. Consequently, the analysis in the previous section does not reflect accurately the achievable throughput of networks with mixed traffic-type sources or remote units with high-order modulations or high bandwidths.

Unfortunately, an exact analysis of the RBM-T protocol throughput under more realistic traffic assumptions is virtually impossible. Furthermore, although the achievable throughput as a function of offered load is of interest, the channel access and transport delays as functions of throughput are perhaps better indicators of the expected real-world protocol performance. Even in the $a \leq 1$ case, analysis of the throughput-delay characteristics is very difficult. Hence, this section presents results of throughput-delay simulations of the $K = 1$ and $K > 1$ protocols under various traffic assumptions. First, the $a \leq 1$ constraint is eliminated. Second, the assumption that the offered traffic is Poisson distributed is removed. Instead, the process of *new message arrivals* is assumed to be Poisson distributed, and the offered load then takes on some unknown distribution. Because the multicarrier system is forced to operate in a TDMA fashion, all messages are still constrained to require an integer number of multicarrier symbols for transmission. Therefore, the multicarrier system is not penalized for supporting a large number of bits per symbol. Finally, the remote terminals are assumed to employ a *nonpersistent* rescheduling algorithm, which means that if the channel is being used for data transmission when the remote checks the channel state, the remote reschedules its request attempt for a later time. In other words, if the channel is in use, a remote unit does not monitor the channel status signal and attempt to transmit its reservation request during the first contention symbol after the current data transmission. Instead, the reservation request is delayed for a time determined by a delay distribution, at which time the channel status signal is checked again and the request transmitted if the channel is idle. A request involved in a collision is rescheduled likewise after the remote unit has determined a collision occurred.

Simulation Methodology

The RBM-T protocol was simulated using Matlab™. In all simulations, consistent with the definition of a single group in the reference system described in Chapter 3, a total bandwidth of 4.416 MHz was assumed. However, the number of subchannels into

which that 4.416 MHz band was divided varied from 8 to 128. A maximum of 1024 remote terminals were assumed to be assigned to the bandwidth considered, which means node addresses of at least 10 bits were required. All remote terminals were assumed to be located exactly 2 miles from the fiber node, which itself was assumed to be 50 miles from the headend. The propagation times of coaxial cable and fiber were taken to be 7.5 μs/mile and 5.5 μs/mile, respectively. Consequently, the one-way propagation time for each remote terminal in all simulations was 290 μs. Although the bit distributions of the remote terminals can in general vary significantly, as Chapter 3 showed, to enable evaluation of the protocol independent of the modulation, all remote terminals were constrained to transmit the same number of bits per symbol. For message transmissions, the average number of bits per subchannel was assumed to be 4 bits. Thus, although the bit distributions vary from terminal to terminal, in the simulations all remote terminals supported $4\bar{N}$ bits per multicarrier symbol, where $\bar{N}$ is the number of subchannels into which the channel is divided. For reservation requests, because the lowest common denominator bit distribution must be used, each subchannel was used to transmit only 2 bits. Hence, the simulations represent the worst-case protocol performance, as the minimum spectral efficiency has been imposed on reservation requests.

Although only 10 bits are required to represent the node addresses of the 1024 remote terminals that could access this group, 16-bit reservation requests were assumed. The additional 6 bits could be used by the remote terminals in a real system to add redundancy for noise immunity, to request bandwidth for the transmission of more than one message, or to indicate service requirements.

The simulations proceeded in the following manner. For each throughput value, ranging from 0.1 to 1.0, a matrix of 100 events was generated. Each row of the matrix represented a reservation request, characterized by the node address of the requesting terminal (generated using a uniform random variable), the arrival time of the reservation request (generated using an exponential random variable), the size of the associated message in bits (which was held constant in these particular simulations), and the current delay of the reservation request in multicarrier symbol periods. The matrix was sorted in order of request arrival times. Because reservation requests were generated according to a continuous-time Poisson process, the delay of *new* reservation requests was set to one-half a multicarrier symbol period to account for arrivals

between multicarrier symbol boundaries. Vectors representing the channel state and the use of frequency-domain request slots were maintained to track use of the channel and reservation request outcomes. Each iteration of the simulations began by updating the channel status vector based on the arrival time of the first request in the event matrix. The first element of the channel status vector was then checked to determine if the channel was available for transmission of reservation requests. If the channel was available, then the first request in the event matrix and any other requests in the matrix scheduled during the same multicarrier symbol period were "transmitted" by generating a uniformly-distributed request slot index between 1 and K for each request. The outcome of each request was then determined by checking whether any other remote terminal chose the same reservation slot. If a request was successful, meaning a reservation slot was chosen by exactly one remote terminal, then, based on the channel status vector and the round-trip propagation time of the network, the indices of the symbols assigned to the remote terminal for transmission of its message corresponding to the successful reservation were determined. The channel status vector was then updated to reflect the assignment. The channel access delay was then computed as the difference between the *initial* arrival time of the reservation request and the time at which transmission of the corresponding message begins. The transport delay was set to the access delay plus the time required to transmit the message, which was the same for all remote terminals because the number of bits transmitted during each symbol period was the same. If a request was unsuccessful, meaning more than one remote terminal transmitted reservation requests on a particular subchannel set, then a rescheduling delay was computed from an exponential random variable. The delay and arrival time of the request were then updated to include the round-trip propagation delay plus the rescheduling delay, and the rescheduled request was inserted, based on its new arrival time, in the appropriate location in the event matrix. If the channel status signal indicated the channel was in use for data transmissions, then any requests scheduled for that symbol period were rescheduled according to an exponential delay distribution. Their arrival times and delays were then updated to reflect the rescheduling delay, and the event matrix was sorted by arrival times.

After each iteration, regardless of whether any successful reservation requests occurred, at least one new reservation request was generated.[6] The times of arrival of

[6]If j successful requests occurred during the symbol period considered, then j new reservation

new reservation requests were computed based on the last *new* request in the event matrix to ensure the desired throughput was maintained during the simulations.

For each throughput $S(i)$, bandwidth was allocated for the transmission of $20,000 + 100,000S(i)$ data messages, all of a constant length, to ensure the protocol had reached steady-state.[7] It was verified that the average delay had stabilized for each throughput value.

Simulation Results

Figure 5.9 shows the expected channel access delay as a function of throughput under the RBM-T protocol. A message length of 512 bits was used, and the number of subchannels was varied from 8 to 128. Given each subchannel was assumed to support 2 bits for reservation requests, the value of K varied from 1 to 16 while the total system bandwidth was held constant. Because the number of subchannels was a variable, it was assumed that each subchannel supported exactly 4 bits for data transmissions so that the spectral efficiency of each system was the same. Thus, when $K = 1$, meaning the channel was partitioned into 8 subchannels, 16 symbols were assigned for each message transmission. When $K = 16$, only one symbol was assigned for transmission of a message.

Figure 5.9 shows that dividing the channel bandwidth into larger numbers of subchannels results in slightly larger access delays as the throughput increases. This result agrees with the throughput analysis in the previous section. The throughput analysis showed that the gains in throughput achieved by higher-order multicarrier systems occur only at very high loads. The simulation results indicate that the access delay at these loads is very high; so high, in fact, that no gain is evident in the throughput-delay curves as K increases.[8] As mentioned previously, the transport delay of each configuration is just the access delay plus a constant that is the same

requests were generated to ensure the event matrix always contained at least 100 new requests.

[7] In actuality, as in Aloha, there is a small but nonzero probability that a reservation request in the RBM-T protocol could collide indefinitely with other requests, which means that a true steady-state condition is never reached. However, it is expected that usage of HFC networks, like Ethernet, will vary significantly with time of day. Overnight, for example, the offered load is likely to be light, which would allow the RBM-T protocol to stabilize if it were overloaded slightly during the day.

[8] This conclusion may not hold in general. There may be channel parameters (that is, round-trip delay, etc.) that would cause the higher-order multicarrier systems to perform better.

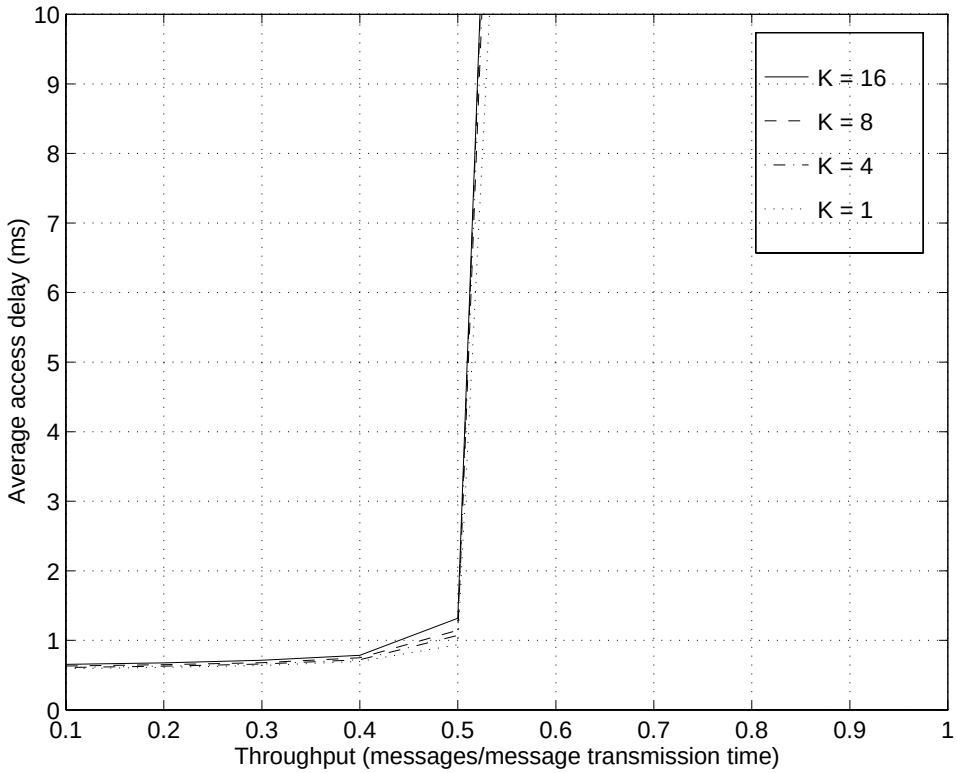

Figure 5.9: Channel access delay versus throughput for RBM-T protocol with $K = 1$ through 16

for any value of K, as shown in Figure 5.10. From the simulation results, it is clear that using a multicarrier system rather than a single-carrier system with the RBM-T protocol yields no improvement in either access or transport delay. This conclusion suggests that constraining a multicarrier system to operate in a TDMA-like mode for data transmission results in a slight degradation in performance with respect to a spectrally-equivalent slotted single-carrier system operating under the same protocol. It is important to note that a multicarrier system will, in general, support more bits per Hertz than a single-carrier system, so there may in fact be a gain when the constraint of equivalent spectral efficiencies is removed.

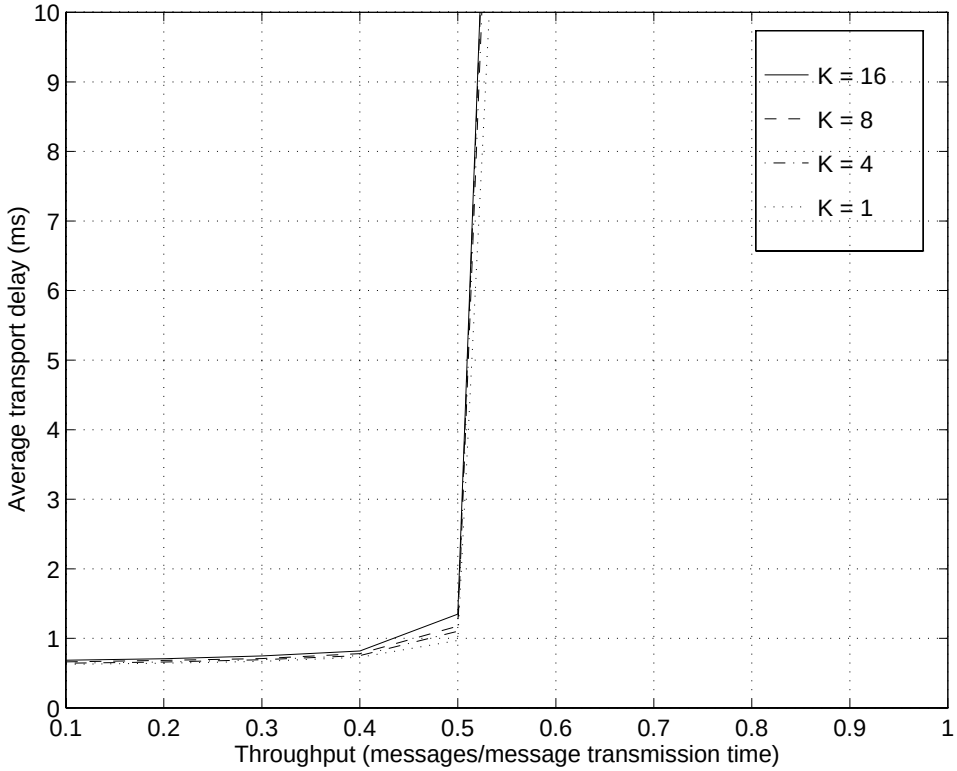

Figure 5.10: Message transport delay versus throughput for RBM-T protocol with $K = 1$ through 16

5.3 The RBM-TF Protocol: A Reservation-Based Multicarrier Protocol with Combined TDMA and FDMA Data Symbol Assignments

The throughput-delay curves presented in the previous section suggest that assigning multicarrier symbols for data transmission in a TDMA fashion does not yield any protocol performance gain. One reason is that the channel is blocked from reservation attempts when data is transmitted; any reservation requests that might otherwise be attempted during that time must be rescheduled. These rescheduled requests are then more likely to collide once the channel is available because the rescheduling delays must be finite in practice to achieve satisfactory access and transport delays. At some throughput value, this effect causes a catastrophic failure of the protocol. This result

motivates the design of another RBM protocol that takes advantage of the dimensionality in both time and frequency of multicarrier modulation by assigning only some of the total number of subchannels during each symbol period to a remote that transmits a message. In this protocol, called the *Reservation-Based Multicarrier protocol with combined TDMA and FDMA data symbol assignments (RBM-TF)*, each symbol is divided by the headend controller into J sets of subchannels for data transmissions. The subchannel sets are called *data slots*. The data slots may be composed of the same number of subchannels, or they may contain different numbers of subchannels. Regardless of what method is used to divide the subchannels into data slots, the partitioning must be observed by all remote terminals. Given a particular data slot partitioning, any of the J sets can be assigned to a particular remote terminal for the number of symbols required to transmit a message. Because the remote terminal bit distributions may vary significantly, the jth data slot may support a variable number of bits per symbol, depending on which remote terminal uses it for data transmission. Hence, a B-bit message transmitted by Remote A on data slot j may require L symbol periods, whereas that same message transmitted by Remote B using the same data slot may require more or fewer symbols, depending on Remote B's bit distribution. The channel status signal, broadcast by the headend controller in the downstream control channel, indicates which of the J transmission slots will be in use for data transmission during the next symbol. Hence, if exactly n message slots will be in use during a particular symbol period, the remaining unused $N - nN/J$ subchannels are partitioned into $K(t)$ request slots, where the maximum value of $K(t)$, which occurs when no data slots are in use during a symbol period, is just K as defined for the RBM-T protocol. Remote units in the ready state can then choose randomly from among the available $K(t)$ request slots and transmit their reservation requests during the next symbol period, sending only two bits per subchannel and applying the inverse FEQ to the reservation requests. As a simple example of the RBM-TF protocol partitioning, if a 128-subchannel multicarrier system that supports exactly 4 bits per subchannel per remote terminal for 512-bit data messages divides the channel into $J = 4$ transmission slots, then a transmission slot consists of 32 4-bit subchannels assigned for 4 consecutive symbols. If 16-bit reservation requests are assumed, then each unused transmission slot during a single symbol period corresponds to 4 request slots. Figure 5.11 shows an example of the channel activity with $K = 4$ and $J = 2$.

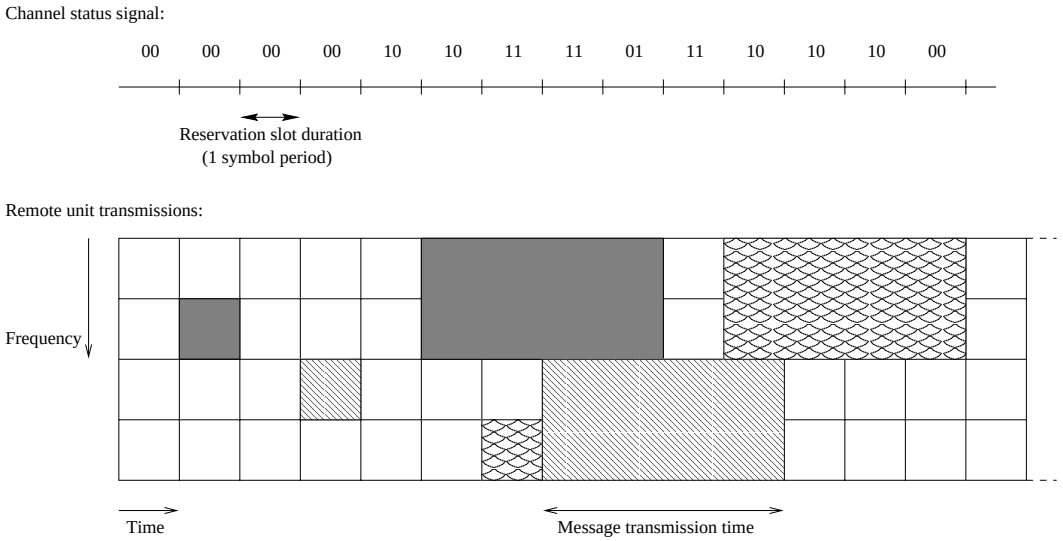

Figure 5.11: Illustration of RBM-TF protocol with $K = 4$ and $J = 2$

As the figure suggests, for throughput values below some maximum, the combined TDMA/FDMA assignment of symbols results in a higher probability that at least one request slot is available during each symbol period. Clearly, as J increases, so does the probability that at least one request slot is available. The penalty for the availability of additional request slots is, of course, an increase in message transmission time incurred by forcing remote units to transmit using only $1/J$ of the total bandwidth.

5.3.1 Simulation Results: Throughput-Delay Curves

Simulation Methodology

Like the RBM-T protocol, the RBM-TF protocol was simulated using Matlab™. Again, 16-bit reservation requests were assumed. As before, the reservation slot outcomes were tracked using a vector. However, whereas the channel status signal in the RBM-T protocol could be represented by a vector, in the RBM-TF protocol maintaining a matrix was necessary to keep track of data slot assignments.

The RBM-T protocol simulation results showed that no throughput-delay advantage results from partitioning the channel into a large number of subchannels. In the

RBM-TF protocol, however, the subchannels are grouped into both data and reservation slots. Hence, to maximize the number of J values into which the subchannels could be partitioned, the simulations used 128 subchannels. Furthermore, whereas in the RBM-T protocol only the *total* number of bits per symbol was constrained to be the same for all remote terminals, in the RBM-TF simulations all subchannels for all remote terminals were assumed to support exactly 4 bits each. This assumption greatly simplified bookkeeping in the simulations, but, as mentioned previously, constraining the subchannels to support the same number of bits is not necessary in a real system as long as all remote terminals observe the same data and reservation slot partitioning. Allowing arbitrary bits per subchannel does increase the burden on the headend controller, however, as the controller must compute per message the number of symbols for which a data slot should be assigned.

Simulation Results

Figure 5.12 shows the average access delay of messages transmitted under the RBM-TF protocol for various values of J. Note that $K = \max[K(t)] = 16$. From Figure 5.12, it is clear that as the number of transmission slots available during each symbol increases, the maximum achievable throughput also increases. Furthermore, the average access delay at any throughput value is smallest for larger J values. Hence, dividing the available transmission bandwidth into J transmission slots can decrease the channel access delay. However, as mentioned previously, the time required to transmit each message increases in direct proportion to J, which means that for each value of J the access delay must be increased by a different amount to compute the transport delay. Figure 5.13 shows the average transport delay of messages for the various values of J. The figure demonstrates that the long message transmission time required for higher values of J significantly increases the transport delay, even at low throughput values. Looking only at Figure 5.12, one might conclude that setting $J = 32$ results in the best system performance (with respect to the other two values of J used in the simulations). However, Figure 5.13 indicates that $J = 32$ may not be the best choice. The average transport delay with $J = 32$ is approximately twice that with $J = 4$ at low loads, and at higher loads $J = 32$ offers only a slight decrease in transport delay over a small range of throughputs. Hence, this example illustrates the importance of considering both access delay and transport delay when choosing

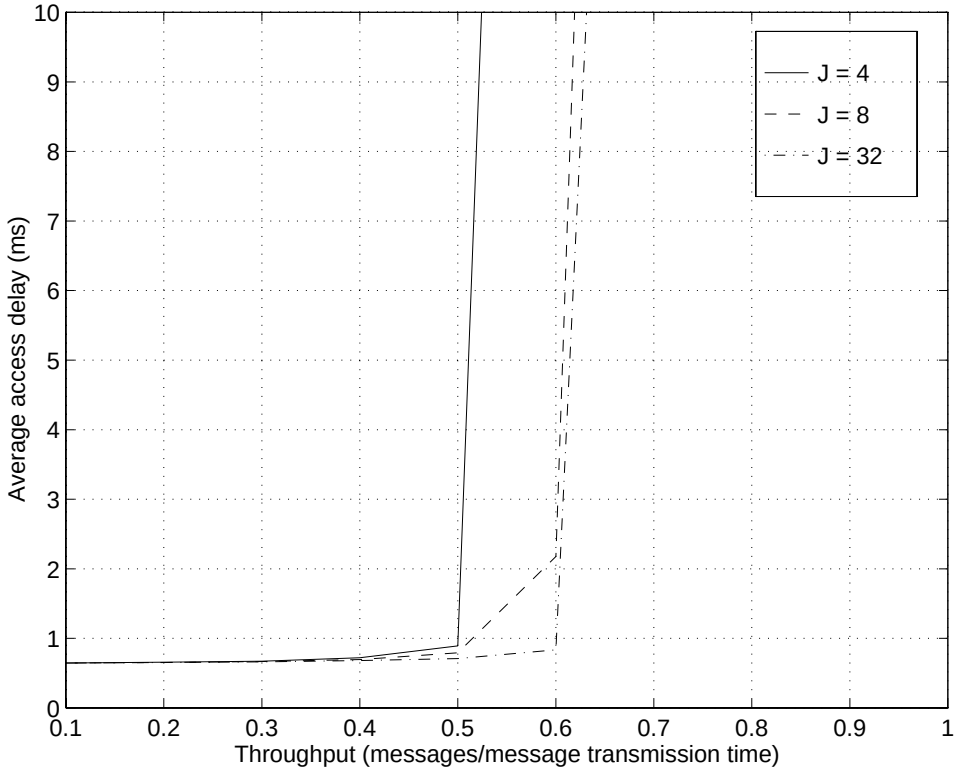

Figure 5.12: Channel access delay versus throughput for RBM-TF protocol with 128 subchannels and $J = 4$, 8, and 32

a multicarrier access protocol configuration.

5.4 Summary

The chapter began by reviewing a number of channel access protocols for use in the HFC reverse channel. TDMA, FDMA, Aloha, CSMA, and BTMA were all discarded because none of them offers an attractive set of performance characteristics for the HFC upstream environment. It was concluded that reservation-based protocols *do* meet the requirements for an HFC reverse channel protocol. Subsequently, new reservation-based protocols designed specifically for multicarrier remote terminals were presented. In particular, the Reservation-Based Multicarrier protocol with TDMA data symbol assignments (RBM-T) was proposed. The performance of the RBM-T protocol, in terms of how well it enables the channel to be used and the

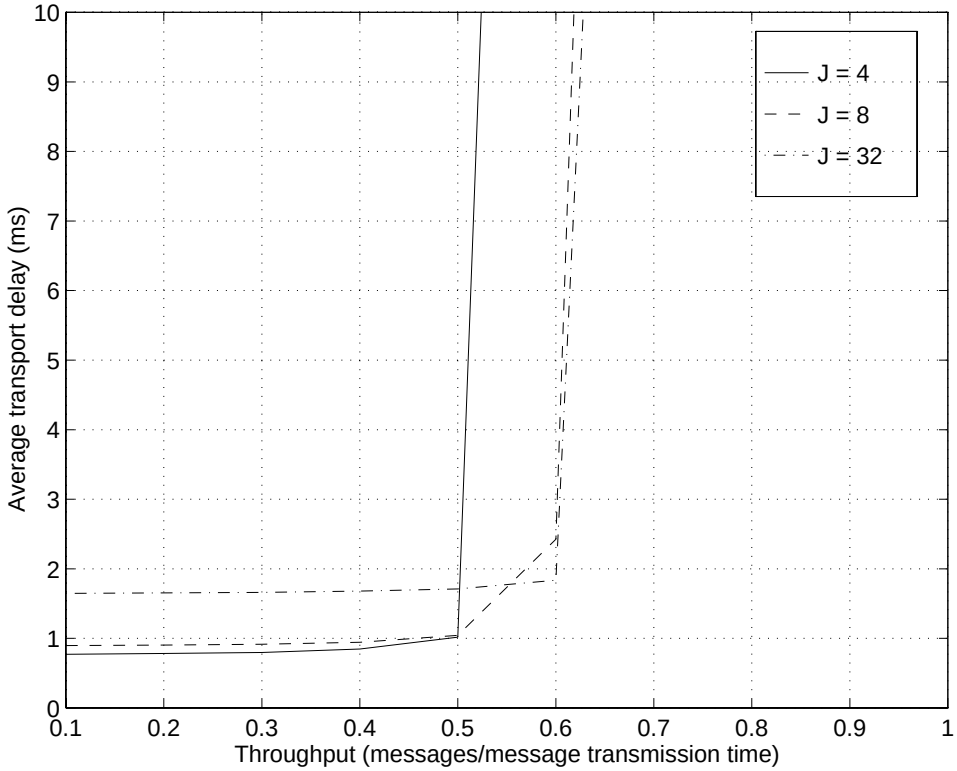

Figure 5.13: Message transport delay versus throughput for RBM-TF protocol with 128 subchannels and $J = 4$, 8, and 32

delays messages incur due to the protocol, was evaluated via analysis and simulation. Because the performance of the RBM-T protocol was not as good as desired, a variation on the RBM-T protocol, called the Reservation-Based Multicarrier protocol with combined TDMA and FDMA data symbol assignments (RBM-TF), was described. Simulation results were presented to illustrate that the RBM-TF protocol outperforms the RBM-T protocol.

Chapter 6

Conclusions

This dissertation began with a statement of the goal of this research: to design a digital communications system for the reverse channel of HFC networks. In addition to summarizing the results, this chapter reviews how well the design presented meets the requirements outlined in Chapter 1.

6.1 Summary of Results

Chapter 2 began with a review of HFC networks. The tree-and-branch architecture in which HFC networks are usually configured was explained, and the limitations this topology imposes on reverse channel transmission were described. Next, common reverse channel impairments were identified. Causes of the most prevalent and severe impairments—micro-reflections, thermal noise, ingress noise, and impulse noise—were described. These impairments were then incorporated into a worst-case equivalent reverse channel model.

In Chapter 3, the first piece of the reverse channel communications system design was described. Because Discrete Multi-Tone offers resilience to the common HFC impairments discussed in Chapter 2, it was proposed as the modulation for reverse channel transmission. The chapter showed that DMT transmits data only in regions of the spectrum in which meaningful communications can be supported. Additionally, the ability of DMT to adapt to a channel's frequency response and noise characteristics was noted as advantageous for reverse channel transmission. The procedure for computing the optimal bit distribution for a transmission channel was then described,

and the procedure was applied to the channel modeled in Chapter 2, thus illustrating the high data rates achievable with DMT. Next, the implementational complexity of DMT was evaluated. Based on this evaluation, it was concluded that a remote terminal design spanning the entire reverse channel is impractical. Hence, a reference system was defined, in which the reverse channel is partitioned into a total of 1024 subchannels. The 1024 subchannels are subsequently divided into eight groups of 128 subchannels to enable the design of cost effective remote terminals.

Information presented in Chapter 3 illustrates that using DMT as the reverse channel modulation meets four of the design requirements outlined in the introduction:

1. **Compatibility with existing HFC network designs, parameters, and services**: By partitioning the reverse channel bandwidth into 1024 34.5-kHz-wide subchannels, the system observes the reverse channel bandwidth allocation between 5 and 42 MHz. Furthermore, if other systems are present in the reverse channel bandwidth, DMT can nimbly avoid bandwidth conflicts simply by not using those subchannels that overlap the bands used by other systems. Hence, compatibility with existing HFC network designs, parameters, and services is assured.

2. **Efficient use of the limited reverse channel bandwidth**: Because DMT allocates data to the subchannels based on their signal-to-noise ratios, it maximizes the system efficiency. Hence, the data-carrying capability of the limited reverse channel bandwidth is fully exploited.

3. **Ability to support up to 2,000 users per fiber node**: By maximizing the data rate supported by the reverse channel, DMT simultaneously maximizes the number of users that can be supported by the network. The results of Chapter 3 indicate that the aggregate data rate is a function of the transmit power used and the noise characteristics of the channel. In addition, the number of users that can be supported at any given time depends on the data rate requirements of the remote terminals. Assuming an 80 dB transmit power-to-AWGN ratio, if 2,000 remote terminals are active, each can transmit at approximately 84 kbps, on average, which is significantly higher than the data rates offered by voice-band modems.

Hence, even when the HFC network is used heavily, the data rates available to the remote terminals are quite high.

4. **Resilience to a variety of channel noises and impairments**: The chapter simulations and discussion illustrated clearly that DMT is robust in the presence of all common HFC reverse channel impairments. Channel nulls and regions of the channel that are corrupted by ingress are avoided by assigning fewer or no bits to the subchannels that overlap those regions of the reverse channel. By adapting the bit distributions via a bit swap algorithm, peak performance is maintained even as the channel and noise characteristics change. DMT's inherent resilience to impulse noise was also described, and methods for improving the system robustness were discussed.

In Chapter 4, the multi-access part of the problem was considered for the first time. Complications that result from using DMT in a multipoint-to-point, rather than point-to-point, environment were discussed. In particular, the loss of the cyclic prefix when remote terminal transmissions are not coordinated was illustrated by an example. The example emphasized the importance of maintaining the cyclic prefix in multipoint-to-point transmission to enable the receiver to eliminate intersymbol interference simply by discarding the cyclic prefix. As a result of this discussion, Synchronized DMT was proposed to enable a set of independent HFC remote terminals to communicate with the headend modem on the reverse channel. It was shown that synchronizing DMT-based remote terminal transmissions preserves the cyclic prefix, and hence the desired subchannel orthogonality. Under the assumption that SDMT is used in the reverse channel communications design, new procedures designed specifically for installing, training, and retraining DMT-based remote terminals were described.

Chapter 4 revealed the design's ability to meet another of the requirements from Chapter 1:

5. **Support of a unique network address at the subscriber premise**: The SDMT installation method presented in Chapter 4 requires installing remote terminals to exchange their factory-assigned serial numbers for node addresses to facilitate more efficient communications between the

headend modem and the remote terminals. Hence, support of a unique network address at the subscriber premise is assured.

Chapter 5 began by reviewing a number of candidate channel access protocols for use on HFC reverse channels. It was concluded that the best choice for the restrictive HFC reverse channel is a reservation-based protocol. Two new reservation-based protocols designed specifically for multicarrier-based remote terminals were then presented. Both protocols use an innovative partitioning of multicarrier symbols into a set of disjoint but concurrent reservation slots. Remote terminals wishing to request bandwidth then transmit a reservation request on a randomly-chosen reservation slot. The Reservation-Based Multicarrier protocol with TDMA data symbol assignments (RBM-T protocol) was described in detail, and its performance in terms of achievable channel throughput was analyzed. Simulation results were then presented to quantify the throughput-delay characteristics of the RBM-T protocol. The somewhat lackluster results motivated the design of a second protocol, called the Reservation-Based Multicarrier protocol with combined TDMA and FDMA data symbol assignments (RBM-TF protocol). Simulation results were then presented to illustrate that the RBM-TF protocol outperforms the RBM-T protocol and achieves satisfactory throughput-delay characteristics.

Chapter 5 illustrates that the reverse channel design proposed in this dissertation meets the remainder of the requirements from Chapter 1:

6. **Ability to support a variety of services**: To support a variety of services, the design must be able to support variable-length packetized messages in addition to constant bit rate traffic. Although the simulation results presented in Chapter 5 assumed all remote terminals transmit only packetized messages of a particular length, this assumption was made to simplify computation of the channel throughput. There is nothing inherent in either the RBM-T protocol or RBM-TF protocol to preclude the transmission of variable-length messages. If the remote terminals specify the sizes of their messages in the corresponding reservation requests, then the headend controller can simply allocate a number of symbols or subchannels sufficient to support the messages. Likewise, both the RBM-T and RBM-TF protocols can support constant bit rate traffic. In either

case, subchannels can be assigned on a "permanent" basis to remote terminals for constant bit rate transmissions. The headend controller can then inform the remote terminals which of the subchannels are in use for constant bit rate transmissions and, hence, unavailable for reservation requests or message transmissions. It is this ability to support seamlessly both packetized and constant bit rate traffic that is one of the strongest arguments for using multicarrier modulation for HFC reverse channel transmission.

7. **Flexible, fluid, and, to the subscriber, transparent allocation of the available spectrum**: Because the RBM-T and RBM-TF protocols enable an elegant allocation of the available spectrum based on remote terminal needs, this requirement is met.

8. **Ability to track subscriber usage of the return channel**: Because all transmissions are scheduled and tracked by the headend controller, remote terminal usage of the reverse channel bandwidth can be tracked easily.

The last design goal, minimizing system cost, was considered in each chapter as decisions were made and system parameters were chosen. At each stage of the development, an attempt was made to minimize the system complexity while still meeting the requirements detailed in Chapter 1.

6.2 Recommendations for Future Research

A variety of topics for future research have been identified as a result of this work. First, there are a number of issues related to the particular protocols described in this dissertation. For example, it was assumed in both the analysis and simulations that a collision between two reservation requests renders both indecipherable by the receiver. In reality, it may be possible to reconstruct one or more requests involved in a collision. If so, then this ability affects the performances of the protocols. Second, this work concluded that reservation-based protocols are a good choice for transmission in HFC reverse channels. However, reservation-based protocols are not the only choice;

many protocol families exist, and surely other protocols that offer good performances in this environment can be found. These new protocols introduce new challenges for installing, training, and retraining multicarrier-based remote terminals. Furthermore, the area of random access protocol design for multicarrier-based remote terminals encompasses many environments in addition to the HFC reverse channel. As an example, protocols for environments in which the remote terminals can sense the state of the channel have been proposed by Chlamtac and Ganz [28] and others, but these protocols only consider multicarrier remote terminals that can transmit on only one subchannel. Perhaps performance enhancements are possible by allowing the remote terminals to transmit using multiple subchannels. Finally, the definition of the "optimal" combination of a modulation scheme and channel access protocol is an open problem.

Appendix A

Coaxial Cable Equations

In this appendix, the calculations required to compute the attenuation of coaxial cable are presented. In particular, expressions for the primary parameters of coaxial cable are given.[7] [29] [30]

Assuming the chosen dielectric material is a good insulator, the attenuation of coaxial cable in terms of its distributed parameters is

$$\alpha = \pi f \sqrt{LC} \left(\sqrt{1 + \left(\frac{R}{2\pi f L} \right)^2} - 1 \right)^{1/2}, \tag{A.1}$$

where L, C, and R are the inductance, capacitance, and resistance, respectively, per unit length of line, and f is the operating frequency in Hz. In turn, L, C, and R are given by

$$L = \frac{\mu}{2\pi} \ln \left(\frac{r_{is}}{r_{oc}} \right), \tag{A.2}$$

$$C = \frac{2\pi\varepsilon}{\ln \left(\frac{r_{is}}{r_{oc}} \right)}, \tag{A.3}$$

$$R = \frac{1}{S_i} + \frac{1}{S_o}, \tag{A.4}$$

where r_{is} is the inner radius of the outer conductor, r_{oc} is the outer radius of the inner conductor, as illustrated by Figure A.1, and μ and ε are the permeability and permittivity, respectively, of the dielectric material. In the equation for R, S_i and

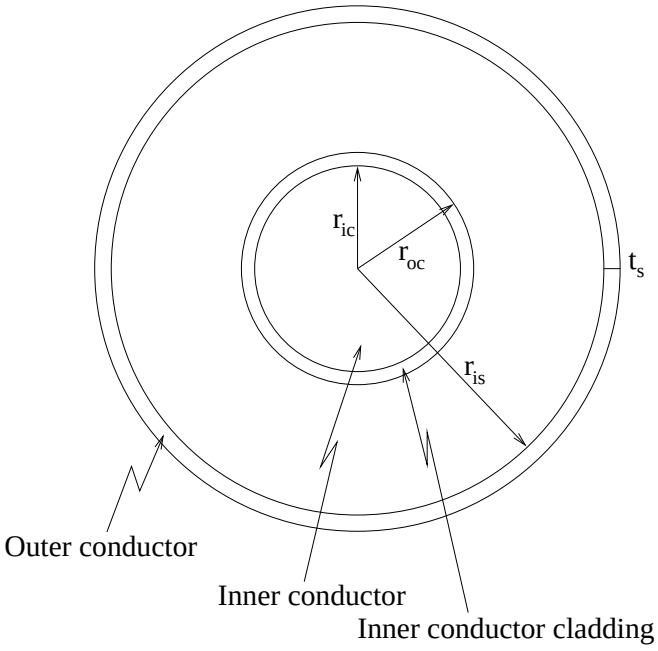

Figure A.1: Coaxial cable cross-section

S_o are the conductances of the inner and outer conductors, respectively. Because the inner conductor is generally either aluminum or steel clad in copper, S_i is a function of the conductances of two metals. For notational simplicity, upcoming equations assume an aluminum center conductor. Additional dimensional variables required in the conductance expressions are shown in Figure A.1: r_{ic} denotes the inner radius of the center conductor copper cladding, and t_s is the outer conductor (or sheath) thickness. The conductivities of aluminum and copper are denoted as σ_{Al} and σ_{Cu}. Using the set of defined variables, S_i and S_o are written as

$$S_i = 2\sigma_{\mathrm{Cu}}\left[\alpha_{\mathrm{Cu}} r_{oc} - \frac{\pi}{\alpha_{\mathrm{Cu}}^2}(\alpha_{\mathrm{Cu}} r_{ic} - 1)e^{-\alpha_{\mathrm{Cu}}(r_{oc}-r_{ic})} - 1\right] + \frac{2\pi\sigma_{\mathrm{Al}}}{\alpha_{\mathrm{Al}}^2} e^{-\alpha_{\mathrm{Cu}}(r_{oc}-r_{ic})}\left[-1 + \alpha_{\mathrm{Al}} r_{ic} + e^{-\alpha_{\mathrm{Al}} r_{ic}}\right] \quad \text{(A.5)}$$

and

$$S_o = \frac{-2\pi\sigma_{\mathrm{Al}}}{\alpha_{\mathrm{Al}}^2}\left[1 - \alpha_{\mathrm{Al}}(r_{is} + t_s) + (\alpha_{\mathrm{Al}} r_{is} - 1)e^{-\alpha_{\mathrm{Al}} t_s}\right], \quad \text{(A.6)}$$

where

$$\alpha_{\mathrm{Al}} = \sqrt{\pi f \mu \sigma_{\mathrm{Al}}}$$

is the attenuation of aluminum at frequency f, and

$$\alpha_{\mathrm{Cu}} = \sqrt{\pi f \mu \sigma_{\mathrm{Cu}}}$$

is the attenuation of copper at frequency f.

Appendix B

Practical System Considerations

This appendix considers some practical issues either deferred or ignored in Chapters 1 through 6 of the dissertation. In Section B.1, the effect of using a cyclic prefix that is not long enough is addressed. An analysis shows that distortion is incurred in the received signal, and a closed-form expression for this distortion is derived. In Section B.2, the effect of a network-wide failure on SDMT is examined. It is concluded that if a small static RAM is included in the remote terminal design, then a network failure is not catastrophic to SDMT. Section B.3 examines the precise memory requirements resulting from the SRAM requirement and the remote terminal and headend memory requirements to use either the RBM-T or RBM-TF protocol. Section B.4 then presents suggestions for reducing the system complexity. The issue of timing accuracy is addressed in Section B.5. A simple example illustrates that the required accuracy in symbol boundary alignment for SDMT is less stringent if the cyclic prefix is longer than the length of the sampled channel impulse response. In Section B.6, requirements for tracking changes in subchannel SNRs are examined. Section B.7 discusses the downstream control channel presumed for both the RBM-T and RBM-TF protocols. The required data rate is computed, and the implications of errors in decoding the downstream control channel are discussed.

B.1 Insufficient Cyclic Prefix Length

As described in Chapter 2, successive DMT symbols are free of ISI if the cyclic prefix spans at least as many samples as are in the channel impulse response. In this section,

the effect of using a cyclic prefix shorter than required is examined. Specifically, the length of the channel impulse response is denoted as $\nu_n+\nu_r+1$, where ν_r is the number of trailing ISI samples, and ν_n is the number of leading ISI samples. The cyclic prefix length is denoted as ν, where $\nu_r + \nu_n > \nu$. Furthermore, the present block of (time-domain) data samples is denoted as $\mathbf{x}$, the previous block as $\tilde{\mathbf{x}}$, the next block as $\mathbf{x}'$, the channel impulse response as $\mathbf{h}$, and the current received block as $\mathbf{y}$. Assuming perfect recontruction of the original discrete-time signal, the current received discrete-time signal can be written as $\mathbf{y} = \hat{\mathbf{x}} * \mathbf{h} + \mathbf{n}$, where $\hat{\mathbf{x}}$ is the concatenation of $\tilde{\mathbf{x}}$, $\mathbf{x}$, and $\mathbf{x}'$, which is necessary because the cyclic prefix is not long enough, and $\mathbf{n}$ is a noise vector of (assumed) IID Gaussian random variables. The number of independent samples in each block is denoted as N_0. Hence, the length of each data block with the cyclic prefix appended is $N = N_0 + \nu$.

To begin the analysis it is assumed that $\nu_n = 0$, which means that all ISI is due to trailing channel impulse response samples. The distortion analysis is simplest if an example for specific values of ν_r, ν, and N_0 is considered first. Generalized results can then be extrapolated for arbitrary values of these variables. Let $\nu_r = 5$, $\nu = 2$, and $N_0 = 7$ to start, and neglect noise for the time being. The present output block can be written as

$$\begin{aligned} y_k &= x_k * h_k \\ &= [\cdots\ \tilde{x}_4\ \tilde{x}_5\ \tilde{x}_6\ \tilde{x}_7\ x_6\ x_7\ x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7\ \cdots] * [h_0\ h_1\ h_2\ h_3\ h_4\ h_5] \end{aligned}$$

After stripping the cyclic prefix, the output samples can be written as

$$\begin{bmatrix} y_1 \\ \vdots \\ y_7 \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & h_2 & h_1 \\ h_1 & h_0 & 0 & 0 & 0 & h_3 & h_2 \\ h_2 & h_1 & h_0 & 0 & 0 & h_4 & h_3 \\ h_3 & h_2 & h_1 & h_0 & 0 & h_5 & h_4 \\ h_4 & h_3 & h_2 & h_1 & h_0 & 0 & h_5 \\ h_5 & h_4 & h_3 & h_2 & h_1 & h_0 & 0 \\ 0 & h_5 & h_4 & h_3 & h_2 & h_1 & h_0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_7 \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 0 & h_5 & h_4 & h_3 \\ 0 & 0 & 0 & 0 & 0 & h_5 & h_4 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \vdots \\ \tilde{x}_7 \end{bmatrix}$$

The first matrix can be written as the sum of two matrices, and the received vector is then

$$\mathbf{y} = H_1\mathbf{x} + H_2\mathbf{x} + H_3\tilde{\mathbf{x}},$$

where

$$H_1 = \begin{bmatrix} h_0 & 0 & h_5 & h_4 & h_3 & h_2 & h_1 \\ h_1 & h_0 & 0 & h_5 & h_4 & h_3 & h_2 \\ h_2 & h_1 & h_0 & 0 & h_5 & h_4 & h_3 \\ h_3 & h_2 & h_1 & h_0 & 0 & h_5 & h_4 \\ h_4 & h_3 & h_2 & h_1 & h_0 & 0 & h_5 \\ h_5 & h_4 & h_3 & h_2 & h_1 & h_0 & 0 \\ 0 & h_5 & h_4 & h_3 & h_2 & h_1 & h_0 \end{bmatrix},$$

$$H_2 = \begin{bmatrix} 0 & 0 & -h_5 & -h_4 & -h_3 & 0 & 0 \\ 0 & 0 & 0 & -h_5 & -h_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & -h_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and

$$H_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & h_5 & h_4 & h_3 \\ 0 & 0 & 0 & 0 & 0 & h_5 & h_4 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

An examination of these matrices reveals that when the cyclic prefix is not long enough the received vector is the sum of three components. The first, the product of a circulant matrix of channel components and the current input block, would be the only contributor to the output if the cyclic prefix length were equal to ν_r. The second and third components are the result of the cyclic prefix *not* being long enough, and they can be considered to be "intrablock" and "interblock" interference, respectively. Hence, as might be expected, if the cyclic prefix is too short, distortion is incurred in the received signal. A variety of metrics can be used to quantify the distortion; the sum of the Frobenius norms of matrices H_2 and H_3 is a convenient measure that is used here. Note that $|H_2| = |H_3|$. For the specific case of $\nu_r = 5$, $\nu = 2$, and $N_0 = 7$, the distortion is

$$|H_2|^2 + |H_3|^2 = 2h_3^2 + 4h_4^2 + 6h_5^2.$$

In the general case, the distortion is

$$D_u = 2 \sum_{i=\nu+1}^{\nu_r} (i - \nu) h_i^2. \tag{B.1}$$

Hence, the distortion due to the cyclic prefix not being long enough is a weighted sum

of squares of the impulse response samples that extend beyond the cyclic prefix. The weights increase linearly with "distance" in samples from the cyclic prefix length.

Although the preceding analysis assumed that all interblock interference is due to trailing samples of the channel impulse response, the analysis can be extended to include precursor ISI samples. Precursor samples interfere with the previous block in the same manner in which post-cursor samples interfere with the next block. Specifically, the precursor channel impulse response samples are weighted in direct proportion to their distance, in samples, from the h_0 sample. If the precursor channel impulse response samples are denoted as $h_{-1}, h_{-2}, \ldots, h_{-\nu_n}$, then the total distortion due to both leading and trailing intersymbol interference is easily derived as

$$D_{u,total} = 2\left[\sum_{i=\nu+1}^{\nu_r}(i-\nu)h_i^2 + \sum_{i=1}^{\nu_n} i h_{-i}^2\right]. \tag{B.2}$$

B.2 Network Failure

One potential situation that may arise during operation of an HFC network is a system failure, in which many or all remote terminals are suddenly unable to communicate with the headend modem. Such a situation could be caused, for example, by a power outage. When power is restored, some method of "restarting" the remote terminals is required.

As was mentioned in Chapter 4, after an SDMT remote terminal has been synchronized, meaning it has loop-timed its local clock to the master clock and acquired its sample delay by transmitting a ranging signal, it does not need to be resynchronized unless it is physically disconnected from the network. Hence, to ensure that a power failure does not necessitate a re-installation of remote terminals, the node address, sample delay, and most recent bit distribution and coarse inverse FEQ tap settings should be stored by the remote terminals in static RAM. Then, following a power outage, the remote units need only loop-time their clocks to the master clock before they may transmit according to the chosen access protocol.

Although the loss of local power does not necessitate resynchronization of a remote terminal, some uncommon but feasible situations may require a remote terminal to repeat the entire installation procedure. For example, if coaxial cable between the remote terminal and the fiber node is inserted or removed for some reason while the

Table B.1: Remote terminal SRAM requirements: reference configuration

Parameter stored	Memory required (in bytes)
Node address	2
Sample delay	2
Bit distribution	48
Inverse FEQ	192
Total	244

unit is operating, then the terminal should be reinstalled to account for the change in cable length between it and the fiber node. In such a case, the remote terminal recognizes the need to reinstall when it detects no energy in the downstream control channel, but local power is intact. Thus, the loss of downstream master clock signal causes the remote terminal to repeat the installation procedure, meaning it resynchronizes and undergoes the training procedure detailed in Chapter 4 to determine the required FEQ tap settings and bit distribution.

B.3 Modem Memory Requirements

Based on the discussion in Section B.2, inclusion of static RAM (SRAM) in the remote terminals is desirable to enable the network to recover quickly from an outage. As mentioned previously, a remote terminal's node address, sample delay, and most recent bit distribution and inverse FEQ tap settings must be maintained in SRAM. Based on the reference system parameters, the storage requirements in bytes are given in Table B.1. In computing the number of bytes required to store the bit distribution, it was assumed that up to 8 bits can be supported on each subchannel. Hence, 384 bits are required to represent the entire bit table. Similarly, the inverse FEQ taps are assumed to be represented by 12 bits each, meaning the settings can be represented by 1536 bits. From the table, it is clear that a 256-byte SRAM is sufficient to store the critical parameters.

The memory requirements in the headend modem are more substantial. If the most efficient operation is desired, then the bit tables and FEQ taps for all remote

terminals on the network must be stored by the headend modem. Assuming a maximum of 8 bits per subchannel, a 32-bit receiver FEQ tap representation, and the reference system parameters, the headend modem must store 640 bytes of information per remote terminal. Assuming the network supports up to 2,000 remote terminals, 1.28 Mbytes of storage is required simply to store the bit distributions and FEQ tap settings of the remote terminals. If the remote terminals are constrained to transmit using identical bit distributions, then the headend modem storage is reduced to just over 1 Mbyte. To ensure that a power failure at the headend does not result in a loss of assigned remote terminal node addresses and other information determined at installation, some parameters should be stored in SRAM or on a hard disc. The channel status and other variables that change dramatically as the network operates should be maintained in ordinary RAM.

In addition to storing the bit distributions and FEQ settings, the headend modem must keep track of which node addresses have been assigned to installed remote terminals. However, if the bit tables and FEQ taps are stored in order, meaning that their storage locations correspond to their node addresses, then the allocated addresses are tracked automatically. Upon receiving an installation signal, the headend controller must only locate an unused row of either the FEQ or bit table matrix to assign an available node address to the installing remote terminal. Another storage requirement is imposed by the use of installation and training periods by the protocols: the headend modem must maintain a counter to determine when installation and training periods occur. Presumably only a few bytes are necessary for this purpose. However, a retraining queue must be maintained for each group, and it must store the node addresses of the active remote terminals in the order in which they are to be retrained. Each retrain queue should be larger than the maximum number of remote terminals that will be assigned to a group to enable inactive remote terminals to be placed at the front of the queue when they are reactivated. Assuming that no more than 500 remote terminals can be assigned to a group, a retrain queue vector with 512 2-byte elements should be sufficient. Thus, for all eight groups, 8192 bytes are required.

In addition to storing the bit distributions and FEQ settings, the headend modem must store the symbol or data slot assignments, depending on which protocol is used. Assuming the headend controller is allowed to allocate symbols up to 1024 symbols in the future and the RBM-T protocol is used, a 1024-element vector with 2 bytes

Table B.2: Headend memory requirements, assuming up to 2,000 remote terminals and reference configuration

Parameter stored	Total memory required	
	RBM-T	RBM-TF ($J = 8$)
Bit tables	256 kbytes	256 kbytes
FEQ tap settings	1024 kbytes	1024 kbytes
Symbol counters	16 bytes	16 bytes
Retrain queues	8192 bytes	8192 bytes
Symbol/slot assignments	16384 bytes	131072 bytes
Total	1.305 Mbytes	1.42 Mbytes
Per remote terminal	652 bytes	710 bytes

per element must be maintained *per group* to keep track of data symbol assignments. Considering all eight groups, 16.4 kbytes are required. If the RBM-TF protocol is used, then the data slot assignment storage requirements increase by a factor of J. Hence, if $J = 8$, and the controller is allowed to allocate symbols up to 1024 symbols in the future, then 8192 bytes per group are required, and 131.1 kbytes are necessary for all eight groups.

Table B.2 details the headend modem memory requirements. The values given are cumulative, not per group. Hence, the memory requirements of the headend modem are significantly higher than the requirements of the remote terminals. However, because one headend modem services up to 2,000 remote terminals, the requirements per remote are reasonable.

B.4 Reducing Remote Terminal Complexities

In Chapter 3, it was determined that partitioning the reverse channel into 1024 34.5-kHz-wide subchannels enables, under most circumstances, the transmission of data rates high enough to support any of the services described in Table 1.1. By dividing the 1024 subchannels into eight groups of 128 subchannels, which is the reference configuration, the maximum remote terminal bandwidth is constrained to 4.416 MHz, which, as discussed in Chapter 3, reduces the computational complexity of the remote terminals. However, for some cost-sensitive applications, it may be desirable to design remote terminals that use fewer than 128 subchannels. For example, a

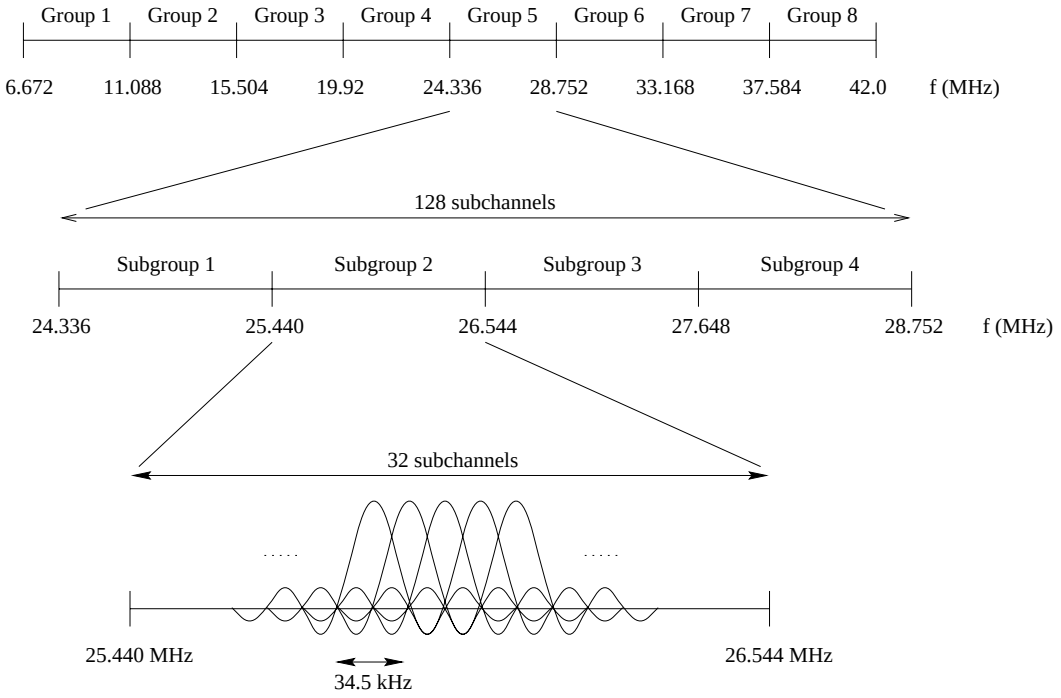

Figure B.1: Partitioning of reverse channel into groups and subgroups

video-conferencing signal may require 384 kbps. Assuming a minimum of 2 bits per subchannel, then at a symbol rate of 32 kbps only 6 subchannels are required to support this bit rate. To guard against subchannels that are too noisy or attenuated to support data, a conservative design for video-conferencing signals might use, say, 32 subchannels. The headend controller could then assign these smaller-bandwidth units to a *subgroup* within a particular group, as shown in Figure B.1. The main advantages of using fewer than 128 subchannels are the resulting reduction in transmitter computational complexity and a reduced digital-to-analog converter complexity. When only 32 subchannels are used by a remote terminal, the required FFT size is 64, and the sampling rate, assuming the reference configuration environment, is 2.208 MHz. To maintain a subchannel spacing of 34.5 kHz, the cyclic prefix length is then 5 samples. Using this configuration, the remote terminal hardware must be capable of executing 24.6 MIPS to compute the FFT at a symbol rate of 32 kHz.

In addition to the computational complexity required to compute the FFTs of data, another contributor to remote terminal cost and complexity is the required

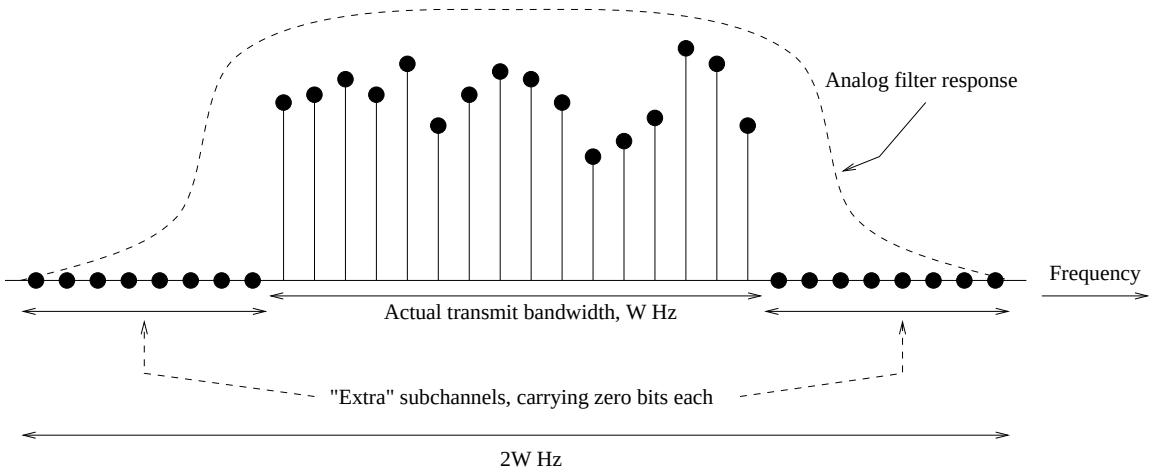

Figure B.2: Use of a $2N$-point FFT to reduce analog complexity: $N = 16$

analog circuitry. Filters are required to confine transmitted signals to the desired group frequency bands. Given the division of the reverse channel into groups of adjacent subchannels, it appears that very sharp transmit filters are required to ensure that signals transmitted by remote terminals in one group do not interfere with signals transmitted by remote terminals in an adjacent group. The cost of these analog filters would likely be prohibitive. To relax the filtering constraints, Bingham has suggested implementing a $2N$-point FFT in the remote terminals and transmitting no energy on the first and last quarters of the N resulting subchannels. [31] Note that the "extra" subchannels on which no energy is transmitted actually belong to the adjacent groups and will be used by remote terminals in those groups, so the "extra" subchannels are not reserved as guard bands. To illustrate the use of a larger FFT than required, in the reference system a 512-point FFT can be used rather than a 256-point FFT. Then less precise, inexpensive analog filters can be designed to ensure only that the transmitted signal attenuation is greater than some minimum value at frequencies beyond the band edges. Figure B.2 illustrates the concept. The penalty for increasing the FFT size is an increase in computational complexity, which was quantified in Chapter 3. However, the cost of complex analog components like a sharp transmit filter, which may be difficult to manufacture reliably, can dominate the total cost of a design, whereas increases in digital complexity are generally not as expensive to implement. Hence, reducing analog complexity is worth the consequent

increase in digital complexity.

B.5 Timing Requirements

The SDMT system design requires that remote terminal transmissions be coordinated so that the symbol boundaries of arriving signals coincide with the symbol boundaries defined by the headend receiver. If the cyclic prefix is not long enough to eliminate ISI between successively transmitted symbols, then, as shown in Section B.1, distortion is incurred in the received signal even if the symbol boundaries are perfectly synchronized. This distortion can become significantly worse if the symbol boundaries are not properly aligned. If the cyclic prefix spans the minimum number of samples required to eliminate ISI, then inaccuracies in symbol boundary alignment at the head end introduce distortion in the received signal in the same manner as when the cyclic prefix is not long enough. However, if the cyclic prefix spans a longer time period than the channel impulse response duration, then the otherwise strict symbol boundary alignment requirements are relaxed somewhat. To illustrate this point, consider a system in which the cyclic prefix is ν samples long, but the sampled channel impulse response is only $\nu_r < \nu$ samples long, and each data block contains N_0 independent data samples. In this case, only the first ν_r samples of each received symbol's cyclic prefix are corrupted by ISI from the previous symbol. Hence, each block of received data, prior to stripping the cyclic prefix, contains $N_0 + \nu - \nu_r$ ISI-free samples. The receiver discards ν samples, of course, but when $\nu > \nu_r$, there are $\nu - \nu_r$ possible sets of data samples that result in ISI-free symbols, as illustrated in Figure B.3. Thus, there may be a discrepancy in symbol boundaries of up to $\nu - \nu_r$ samples before any degradation occurs. If a symbol boundary within the *uncorrupted* cyclic prefix samples is used by the receiver, then the received symbol is shifted in phase with respect to the transmitted symbol (disregarding the additional phase shift introduced by the channel). This phase shift is easily accommodated by the FEQ and causes no degradation to the received signal as long as the symbol boundaries remain constant as the system operates.

To illustrate this concept, in the reference system, every set of 256 (real) data samples is preceded by a 20-sample cyclic prefix. Because the sampling rate of the system is 8.832 MHz, samples are spaced at 113.2 ns intervals, and the cyclic prefix

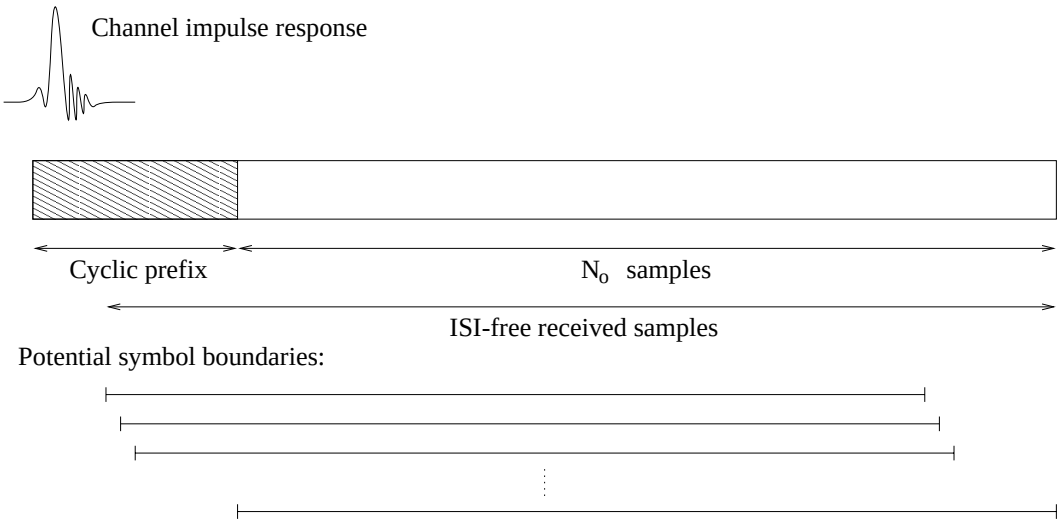

Figure B.3: Illustration of symbol boundary flexibility afforded when $\nu > \nu_r$

therefore spans 2.26 μs. Hence, if the channel impulse response duration is, for example, only 1.5 μs, then only the first 14 samples of the cyclic prefix are corrupted by ISI. Consequently, any contiguous block of 256 samples that begins in the range from sample 15 to sample 20 yields an ISI-free DMT symbol. Therefore, remote unit sample delays need only be within 6 samples of their exact values.

The second component to timing in an SDMT system is carrier alignment. Loop-timing the SDMT remote terminals to the master clock ensures that the carrier frequencies of the remote terminals are precise. In the remote terminal transmitter, the required carrier frequency, which is dependent upon which group of subchannels the remote uses, can be derived easily from the received master clock. Of course, the remote terminal oscillator must be very accurate and stable to ensure the remote clock does not wander.

B.6 Tracking Changes in Subchannel SNRs

As an HFC network operates, both the channel and noise characteristics can change. Damage to a drop cable, for example, can introduce additional nulls in the channel frequency response. The remote terminals and headend modem must adapt to these

Table B.3: Average time between remote terminal retrains: reference configuration

% overhead	# of active terminals/group	Time between retrains (sec)
0.10	10	6.24
0.10	32	19.98
0.10	64	39.96
0.10	128	79.92
0.25	10	2.49
0.25	32	7.98
0.25	64	15.96
0.25	128	31.92
1.00	10	0.62
1.00	32	1.98
1.00	64	3.96
1.00	128	7.92

changes to ensure the system is not disabled as a result of changes in the transmission environment.

In Chapter 4, a procedure for retraining remote terminals was presented. Essentially, a queue is maintained by the headend controller, and active remotes are retrained in a round-robin fashion to keep their bit distributions and required FEQ tap settings up to date.[1] The frequency with which the remote terminals are retrained depends on how many active terminals are on the network and how often the training intervals occur. Table B.3 gives some examples of how often a particular remote terminal can be retrained assuming a certain number of active remote terminals and particular training interval frequencies. For simplicity, the number of symbols per training interval is fixed at 20.

Table B.3 suggests that it might be desirable to allow the headend controller to vary the frequency of training intervals based on the number of active remotes the network is supporting and the number of changes it makes to the various bit distributions as the network operates. For example, if the headend controller notes that significant changes in an active remote terminal's bit distribution are necessary

[1] An exception to the round-robin ordering is a remote terminal that has been off or inactive for a significant time. Upon reactivating, the remote terminal notifies the headend controller that it has become active, and the headend controller moves it to the front of the retrain queue.

during a few consecutive training periods, then it should conclude that the terminals are not being retrained often enough. To remedy the situation, the controller can decrease the time between training intervals, thus increasing the overhead for training.

B.7 Downstream Control Channel

Throughout this dissertation, the availability of a reliable downstream control channel to facilitate reverse channel communications has been assumed tacitly. In part, this assumption is valid: downstream bandwidth in HFC networks is plentiful (or at least more plentiful than the reverse channel bandwidth), and reserving part of this bandwidth for overhead to enable efficient reverse channel communications is worthwhile to cable operators. The assumption that the downstream control channel is reliable may be a less realistic assumption in some cases. Although the downstream bandwidth is generally degraded much less than the reverse channel by transmission impairments, if remote terminals make detection errors when they demodulate the control data, the performance of the reverse channel can be degraded significantly. For example, in both the RBM-T and RBM-TF protocols, the downstream control channel informs the remote terminals of the status of the channel during upcoming symbol periods. Each symbol is classified by its type: installation, training, or other. To ensure collision-free data transmissions and to maximize use of the reverse channel, it is important that the symbol classification signal be decoded correctly by the remote terminals.

The reliability of the downstream control channel data is determined in part by the data rate it supports. Assuming the maximum data rate a fixed-bandwidth downstream control channel can support is known, transmitting at that rate may result in unwanted detection errors by the remote terminals over time, especially if transient channel disturbances such as impulse or ingress noise are expected, or if a decision-feedback equalizer (DFE) is used by the remote terminals to decode the downstream channel.[2] Hence, to attempt to ensure accurate control channel decoding, a noise margin and error correction should be used in the downstream control channel. Assuming a fixed data rate is required to send the downstream control information,

[2]As is well-known, the performance of decision-feedback equalizers can be degraded significantly by error propagation.

imposing a noise margin increases the required transmission bandwidth of the control channel because fewer bits per symbol than the maximum possible are transmitted. To compensate, more symbols must be transmitted per second, which requires an increase in transmit bandwidth.

The precise data rate that must be supported by the downstream control channel depends on whether the RBM-T or RBM-TF protocol is used. In either case, because SDMT requires the remote terminals to be loop-timed to a master clock, the master clock must be broadcast downstream to enable synchronization. The precise frequency of the master clock is not particularly important; the remote terminals can derive the required clock via division. As an example, as an upcoming discussion of the downstream control channel data requirements will show, a 32-kHz clock signal is convenient for the headend modem to transmit and for the remote terminals to use: sampling clocks at 8.832 MHz for those remote terminals that operate using up to 128 subchannels can then be generated by dividing the received master clock by 276. Likewise, remote terminals that operate only within a 32-subchannel subgroup can derive their required 2.204-MHz clocks by dividing the master clock by 69. Specific additional downstream control channel data rate requirements particular to the RBM-T and RBM-TF protocols are described in forthcoming subsections.

B.7.1 RBM-T Downstream Control Channel Requirements and Implications

Consider first the RBM-T protocol. Data symbols are assigned to distinct remote terminals, which means that no reservation requests can be transmitted during data symbols. Hence, the RBM-T symbols can be classified into one of four categories: installation, training, contention, or reserved for data. Perhaps the simplest way to communicate the classification of each upcoming symbol is to encode and transmit downstream a two-bit signal at the upstream symbol rate of 32-kHz. Table B.4 gives an example two-bit constellation mapping that could be used to designate the symbol classifications. If an error is made in decoding the symbol classification signal, the repercussions of the error depend on the particular misclassification. A summary of decoding errors and their general effects is given in Table B.5. The ramifications of specific misclassifications are detailed below.

Table B.4: Two-bit symbol classification signal in downstream control channel

Bit pattern	Symbol classification
00	Installation symbol
01	Training (or retraining) symbol
10	Contention symbol
11	Reserved data symbol

Misclassifying Installation Symbols

As implied by Table B.5, if an installation symbol is incorrectly classified as any other type of symbol by an *installing* remote terminal, then no detrimental outcome occurs: the installing remote simply remains silent and waits for the next installation period. If an already-installed remote terminal incorrectly decodes an installation symbol as a training symbol, then most likely nothing detrimental occurs. As discussed in Section B.6, remote terminals are retrained in a round-robin fashion. The most straightforward way for the headend controller to instruct the appropriate remote terminal to retrain is to send downstream the node address of that terminal prior to the arrival of the next training interval. Thus, unless the remote terminal that misclassified an installation symbol as a training symbol also happens to be the remote terminal next in the retraining queue or also incorrectly decodes its own address in the retraining signal, the remote terminal remains silent. If by some chance the remote terminal *does* transmit a retraining signal during an installation symbol, then the headend controller recognizes that the signal format is not correct, and it broadcasts a message downstream that a problem occurred. Likewise, if an installation symbol is mistaken as a contention symbol by an already-installed remote terminal, then the remote terminal may transmit a reservation request during the installation period. In this case, because installation signals are transmitted via differential QPSK (DQPSK) during two consecutive symbols whereas reservation requests are transmitted during only one symbol period, the headend receiver, which switches to decoding DQPSK signals for the duration of the installation period, is unable to decode the received signal. Consequently, the headend controller recognizes that a non-installation signal was transmitted, and it broadcasts a message downstream to indicate that it was unable to decode the signal(s) transmitted during that

symbol period. If an installation symbol is incorrectly classified as a data symbol, then all remote terminals remain quiet because they are restricted from transmitting during data symbol periods unless authorized by the headend controller. Hence, no performance degradation occurs other than a potential delay in the installation time of an uninstalled remote terminal that misclassified the symbol.

Misclassifying Training Symbols

If a training symbol is decoded as an installation symbol, then an uninstalled remote terminal may transmit an interfering installation signal during the training period. As a consequence, the remote terminal to which the training period is allocated may need to use an additional training period to ensure that accurate bit distributions and FEQ taps are maintained. Alternatively, if the remote terminal that misclassified the symbol is the remote to which that training period has been assigned, the remote will not retrain. The headend controller then detects that the remote terminal has not transmitted its request signal, and it allocates an upcoming training period to the remote terminal to attempt retraining again. Likewise, if a training symbol is mistaken for a contention symbol, training may be corrupted by the transmission of a reservation request, in which case allocation of an additional training period may be necessary. Finally, if a training symbol is mistaken for a data symbol, all remote terminals remain quiet.

Misclassifying Data Symbols

If a data symbol is incorrectly classified as an installation symbol, and if an installing remote terminal consequently transmits an installation signal, then interference with data signals will occur. Because installation signals are two symbols in duration, and they are not synchronized to the symbol boundaries, an unauthorized installation signal could interfere with as many as three data symbols. In this case, the headend controller would likely be unable to decode correctly the data signals, and it would have to allocate future data symbols to the remote(s) that suffered from interference. The effect of mistaking a data symbol for an installation symbol is thus a potential decrease in throughput, since effectively twice the required number of symbol periods

may have to be allocated to transmit a message. Similarly, if a data symbol is mistaken for a training symbol by the remote terminal that is scheduled to retrain during the next training interval, then the training signals interfere with data. Again, the interference could extend over multiple symbol periods, depending on the number of symbols used for training, and the headend must reallocate additional symbols for retransmissions of the corrupted messages. Hence, the protocol throughput may be reduced. If a data symbol is mistaken for a contention symbol, and if the remote terminal that incorrectly decoded the classification signal transmits a reservation request during that symbol period, then the request interferes with some or all of the message transmitted by the remote unit to which that symbol period is allocated. Depending on the error correction capability of the head-end receiver, it may be possible to recover from one or more unauthorized reservation requests. If not, however, then the throughput decreases as additional symbol periods must be allocated to the remote terminal that attempted to transmit data.

Misclassifying Contention Symbols

If a contention symbol is mistaken for an installation symbol, and if an installing remote terminal consequently transmits an installation signal, then some or all reservation requests transmitted during that symbol period may be corrupted by the installation signal. If so, then the headend controller detects reservation request collisions in the received data, and it proceeds as usual after a contention symbol by transmitting a message in the downstream control channel indicating those subchannel sets on which a conflict occurred. If a contention symbol is misclassified as a training symbol by the remote terminal that is scheduled to retrain during the next training interval, then, as in the other cases when this type of mistake is made, the headend controller recognizes that the signal format is not correct, and it broadcasts a message downstream that a problem occurred. If a contention symbol is mistaken for a data symbol, then a reduction in throughput may occur if and only if the remote terminal that incorrectly decoded the control signal would otherwise have transmitted a reservation request during that symbol period. Of course, it is possible that the misinterpretation by the remote terminal might actually improve the throughput if the transmission of its reservation request during that symbol period would have collided with another remote terminal's request. In any case, the effect on system

performance of mistaking a contention symbol period for a reserved symbol period is minimal.

Downstream Control Channel Data Rate Requirements

As described previously, the downstream control channel must support the symbol classification signal (called the channel status signal in the analysis in Chapter 5), which can be implemented as a four-bit signal transmitted at a 32-kHz symbol rate. In addition, the master clock must be supported. However, the clock can be derived easily from data broadcast downstream at, say, 32 ksymbols per second. Thus, broadcasting a separate clock signal is not necessary. The index of the group of 128 subchannels the next installing remote terminal should use must also be transmitted at least once between installation periods. Assuming 0.1% overhead is allowed for installation periods, communicating the group index requires 3 bits sent at least once every 2997 symbols, or at a symbol rate of 11 Hz. Furthermore, the headend controller sometimes must report to the remote terminals that a collision occurred during a symbol period. In general, these messages must be sent to the entire population because the headend controller cannot determine which remote terminals transmitted the requests that collided.[3] When the headend communicates with the entire population of remote terminals, the remote terminals act based on whether they transmitted information on the affected subchannel set during the affected symbol period. Hence, if each contention symbol is partitioned into K reservation slots, then the signal from the headend must be composed of K bits at a 32-kHz symbol rate to indicate whether or not the headend detected a collision on each of the subchannel sets. Finally, the downstream control channel must support the various messages the headend controller sends to instruct particular remote terminals when to transmit data and training signals. Each such headend transmission must include the node address of the remote terminal the controller wishes to address plus the instruction itself. Assuming a 42-MHz reverse channel supports up to 2048 remote terminals, node addresses may need to be up to 11 bits long. Messages addressed to particular remote terminals include instructions to transmit a training signal during the next

[3] In some cases, such as when a collision is detected during a data symbol, the headend controller can deduce the identity of one remote involved in the collision, but, because the other transmission during that symbol period was unauthorized, it cannot determine the address of the interferer.

training interval and indices of symbols assigned for data transmissions. Assuming the headend controller can allocate data symbols up to 1024 symbols in the future, 10 bits are required to indicate a single symbol assignment. A training message, which could be as short as one or two bits in addition to the remote terminal's node address, must be sent at least once between training periods. Assuming 1% overhead is reserved for training periods, then a symbol rate of 17 Hz is required to ensure the next remote in the training queue is adequately informed of its assignment of the upcoming training period. After a remote terminal has transmitted its training signal, both the coarse inverse FEQ tap values and any changes in the bit distribution must be sent back to the remote terminal. In the worst case, the entire bit distribution and an entirely new set of inverse FEQ taps must be sent. To maintain the best system performance, the new settings should be transmitted to the remote terminal before the next training interval arrives. Assuming a 128-entry bit table with up to 8 bits per subchannel, 384 bits are required to represent the bit table. If the inverse FEQ taps can be represented by 12 bits, then a total of up to 1920 bits must be transmitted after each training interval to update the remote terminals. These bits must be transmitted by the time the next training interval arrives, which is 1980 symbols later if 1% overhead for training symbols is assumed. Hence, using a 32 kHz symbol rate, one bit per symbol must be transmitted to send the updated bit table and inverse FEQ settings. Of course, sending the new bit table and inverse FEQ values faster than required may improve the system performance. Thus, although transmitting two or more bits per symbol increases the instantaneous control channel data rate, it may be worthwhile from a performance standpoint. For the purpose of example, it is assumed that 4 bits per symbol are used to transmit the updated values.

Table B.6 summarizes the various downstream control channel signals required *per group* and their required symbol rates and bit rates. From the values in the table, the total required bit rate of the downstream control channel for a single group using the RBM-T protocol is on the order of 1.4 Mbps when $K = 16$. Hence, to control all eight groups, a downstream channel capable of supporting at least 11 Mbps is required. Assuming the assigned downstream spectrum can support at least 6 bits/s/Hz, a 6.0 dB noise margin is desired, and a single-carrier modulation such as QAM is used, the minimum required bandwidth of the control channel is approximately 2.75 MHz, excluding excess bandwidth, since only 4 bits are supported per symbol period.

Assuming significant redundancy is used to represent all downstream control signals, a 6-MHz-wide band of the downstream spectrum should be sufficient to control reverse channel transmissions.

B.7.2 RBM-TF Downstream Control Channel Requirements and Implications

As with the RBM-T protocol, the RBM-TF protocol performance can be degraded when symbol types are misinterpreted by the remote terminals. Under this protocol, however, symbols cannot be classified easily into four categories. Installation and training symbols are the same as in the RBM-T protocol, but "data symbols" and "contention symbols" no longer exist because of the combined TDMA/FDMA data assignments used in the protocol. Instead, every non-installation and non-training symbol period can be described by a J-bit signal called the *data/contention symbol status signal*, where bits set to zero indicate which of the J data slots are in use, and bits set to one indicate those data slots that can be further partitioned into $K(t)$ request slots and used to transmit reservation requests. These slots can be considered contention slots. Hence, symbols can be classified as installation, training, or data/contention by a two-bit signal, but an additional J-bit downstream control signal is required to specify the allocation status of each data/contention symbol. As an example, if a version of the RBM-TF protocol uses $J = 4$ data slots, each of which can be decomposed into 2 reservation slots, then a data/contention symbol with the bit pattern 0100 indicates that the first, third, and fourth data slots are in use for data transmissions, and remote terminals needing to transmit reservation requests during that symbol period should choose one of the two reservation slots into which the second data slot, which is now actually a contention slot, is divided.

The implications of mistaking installation and training symbols for data/contention symbols are the same as in the RBM-T protocol, and Table B.7 describes the effects of misclassifying data/contention symbols. Because the data/contention symbols are partitioned into J data slots, the severity of some misclassifications is mitigated somewhat with respect to the RBM-T protocol. In particular, if a data slot is mistaken for a contention slot, then interference occurs if and only if the remote terminal that made the error needs to transmit a reservation request during that symbol period *and*

also happens to choose one of the reservation slots into which the misclassified data slot is partitioned. Hence, problems due to this type of misclassification are likely to be more frequent at high throughput values.

Downstream Control Channel Data Rate Requirements

As with the RBM-T protocol, the downstream control channel for the RBM-TF protocol must support the symbol classification signal (which can again be implemented as a two-bit signal transmitted at a 32-kHz symbol rate), the data/contention symbol status signal, the group index for the next installing remote terminal, collision reports, training instructions, and data slot assignments.. Of these signals, only the transmission of the data/contention symbol status signal and the data slot assignment signal differs from the RBM-T protocol. If each data/contention symbol is partitioned into J data slots, then J bits must be transmitted at a rate of 32 ksymbols per second to communicate this information to the remote terminals. Likewise, data slot assignments require $\lceil \log_2 J \rceil + 21$ bits, under the same assumptions made for the RBM-T protocol. Table B.8 summarizes the various downstream control channel signals required *per group* under the RBM-TF protocol and their required symbol rates and bit rates. Assuming as an example that $J = 8$ and $\max[K(t)] = 16$, Table B.8 indicates that the total required downstream control channel bit rate for a single group under the RBM-TF protocol is 1.7 Mbps. For all eight groups, the required data rate is 13.8 Mbps. Under the same redundancy and noise margin assumptions as were made for the RBM-T protocol, a 7-MHz downstream band capable of supporting 4 bits per symbol should be adequate to support the required RBM-TF control information.

B.8 Summary

In this appendix, several practical issues related to the system design that were either ignored or deferred in previous chapters were considered. First, the effect of using a cyclic prefix that is not long enough was investigated. The analysis showed that distortion is incurred in the received signal in this case, and a closed-form expression for this distortion was derived. Next, the effect of a network-wide failure on SDMT was examined, and it was concluded that if a small static RAM is included in the remote terminal design, then network failures are not catastrophic to SDMT. The

precise memory requirements resulting from the SRAM requirement and the remote terminal and headend memory requirements to use either the RBM-T or RBM-TF protocol were described next. It was determined that each remote terminal requires approximately 256 bytes of storage, and the headend modem needs less than 1.5 Mbytes. It was concluded that both memory requirements are reasonable. Suggestions for reducing the system complexity were then presented, including allowing the remote terminals to use a maximum number of subchannels that is fewer than 128. In addition, a method for reducing the analog filtering requirements by increasing the FFT size was described. The issue of timing accuracy was addressed next, and a simple example illustrated that the required accuracy in symbol boundary alignment for SDMT is less stringent if the cyclic prefix is longer than the length of the sampled channel impulse response. Requirements for tracking changes in subchannel SNRs were examined next, and it was concluded that the headend controller should be allowed to modify the percent overhead allocated for retraining remote terminals to enable the best possible system performance. Finally, the downstream control channel requirements for both the RBM-T and RBM-TF protocols were addressed. The required control channel data rate was computed for both protocols, and the implications of remote terminal errors in decoding the downstream control channel were discussed. It was concluded that the centralized control required by the RBM-T and RBM-TF protocols mitigates many problems that might otherwise be caused by decoding errors.

Table B.5: Effects of downstream control channel decoding errors by remote terminals under RBM-T protocol

Actual classification	Misclassification	Effect
Installation	Training	Nothing, unless remote terminal that made error is next in retrain queue. An installation signal that might otherwise have been received successfully may have to be retransmitted.
Installation	Contention	Nothing, unless remote terminal transmits reservation request. An installation signal that might otherwise have been received successfully may have to be retransmitted.
Installation	Data	Nothing.
Training	Installation	An installing remote terminal may transmit its installation signal. Remote that was supposed to be training may need an additional period as a result of the interference.
Training	Contention	Nothing, unless remote terminal sends reservation request. Remote that was supposed to be training may need to use an additional training period as a result of the interference.
Training	Data	Nothing.
Data	Installation	An installing remote terminal may transmit an installation signal that interferes with a data transmission.
Data	Training	Nothing, unless remote terminal that made error is next in retrain queue.
Data	Contention	Remote terminal may send a reservation request, which will interfere with a data transmission.
Contention	Installation	An installing remote terminal may transmit its installation signal. A remote that sent a reservation request may need to retransmit the request.
Contention	Training	Nothing, unless remote terminal that made error is next in retrain queue.
Contention	Data	Nothing.

Table B.6: Downstream control channel signals required per group for RBM-T protocol

Signal	Bits required per symbol	Symbol rate	Bit rate
Symbol classification	2	32 kHz	64 kbps
Group index	3	11 Hz	33 bps
Collision detected	K	32 kHz	$32K$ kbps
Data symbol assignment	21	$\leq$ 32 kHz	$\leq$ 672 kbps
Training authorization	13	17 Hz	221 bps
Bit table, inverse FEQ	4	32 kHz	128 kbps

Table B.7: Effects of downstream control channel decoding errors by remote terminals under RBM-TF protocol

Actual classification	Misclassification	Effect
Data/contention symbol	Installation symbol	An installing remote terminal may transmit an installation signal that interferes with a data transmission.
Data/contention symbol	Training symbol	Nothing, unless remote terminal that made error is next in retrain queue.
Data slot	Contention slot	Remote terminals may send reservation requests that interfere with a data transmission.
Contention slot	Data slot	Nothing.

Table B.8: Downstream control channel signals required per group for RBM-TF protocol

Signal	Bits required per symbol	Symbol rate	Bit rate
Symbol classification	2	32 kHz	64 kbps
Data/contention symbol status signal	J	32 kHz	$32J$ kbps
Group index	3	11 Hz	33 bps
Collision detected	$\max[K(t)]$	32 kHz	32 ($\max[K(t)]$) kbps
Data slot assignments	$\lceil \log_2 J \rceil + 21$	$\leq$ 32 kHz	$\leq 32(\lceil \log_2 J \rceil + 21)$ kbps
Training authorization	13	17 Hz	221 bps
Bit table, inverse FEQ	4	32 kHz	128 kbps

Appendix C

Distributing Reservation Requests Among Subchannel Sets

This appendix shows why there are $\begin{pmatrix} n+K-1 \\ n \end{pmatrix}$ ways to distribute n requests among K subchannel sets. The problem is the same as the problem of distributing n balls in K urns, and it can be solved in the following way: Construct a vertical grid of $n+K$ squares. Each square is to be filled by either a ball or an urn, with the restriction that the bottom square must contain an urn. Then, after distribution of the remaining $K-1$ urns and n balls, those balls between two urns can be considered "in" the urn just below them. Hence, after distribution of the balls and urns, the grid might look like Figure C.1, where 6 balls, denoted by circles, are distributed among 5 urns. Beginning from the bottom urn, the urns contain 2, 1, 0, 3, and 0 balls. Given the restriction that the bottom square of the grid must contain an urn, it is clear that after the $K-1$ remaining urns have been distributed among the $N+k-1$ squares, the locations of the balls in the grid are fixed. Therefore, there are $\begin{pmatrix} n+K-1 \\ K-1 \end{pmatrix}$ or, equivalently, $\begin{pmatrix} n+K-1 \\ n \end{pmatrix}$ ways to distribute n balls among K urns.

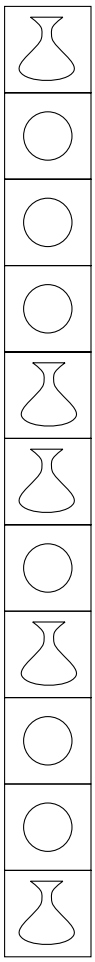

Figure C.1: Distribution of balls and urns on a grid

Bibliography

[1] J.M. Cioffi. "Discrete Multitone Data Transmission System Using an Overhead Bus for Synchronizing Multiple Remote Units". United States patent application filed June 2, 1994.

[2] E. Hare and R. Schetgen. *Radio Frequency Interference: How to Find It and Fix It.* American Radio Relay League, Newington, CT, 1991.

[3] Cable Television Laboratories. "High-Speed Cable Data Service (HSCDS) RFP". Cable Television Laboratories, Boulder, CO, April 1995.

[4] W.S. Ciciora. "An Overview of Cable Television in the United States". Cable Television Laboratories, Inc., Boulder, CO, 1990.

[5] J. Maynard. *Cable Television.* Sheridan House, London, 1985.

[6] Cable Television Laboratories. "Request for Proposals for a Telecommunicataions Delivery System Over a Hybrid Fiber/Coax (HFC) Architecture". Cable Television Laboratories, Boulder, CO, July 1994.

[7] L.V. Blake. *Transmission Lines and Waveguides.* John Wiley & Sons, Inc., New York, 1969.

[8] J.N. Slater. *Cable Television Technology.* Ellis Horwood Limited, New York, 1988.

[9] IEEE 802.14 Working Group. "Functional Requirements Document". In *IEEE 802.14 Document number 94-002R3*, November 1995.

[10] R.S. Prodan, M. Chelehmal, T.H. Williams, and C.M. Chamberlain. "Analysis of Cable System Digital Transmission Characteristics". In *Technical Papers, 43rd Annual National Cable Television Association (NCTA) Convention and Exposition*, New Orleans, LA, May 1994.

[11] R. Pience. private communication. February 1994.

[12] Rogers Cable Systems. "Two-way Plant Characterization". 1994.

[13] R. Citta and D. Mutzabaugh. "Two-way Cable Plant Characteristics". In *IEEE 802.6 Committee Contribution number 94/010*, 1994.

[14] J.A.C. Bingham. "Multicarrier Modulation for Data Transmission : An Idea whose Time Has Come". *IEEE Communications Magazine*, 28(5):5–14, May 1990.

[15] J.M. Cioffi. "A Multicarrier Primer". In *ANSI T1E1.4 Committee Contribution 91-157*, Boca Raton, FL, November 1991.

[16] J. T. Aslanis. *Coding for Communication Channels with Memory*. PhD thesis, Stanford University, November 1989.

[17] S. Kasturia. *Vector Coding for Digital Communication on Spectrally Shaped Channels*. PhD thesis, Stanford University, March 1989.

[18] J.C. Tu. *Theory, Design and Application of Multichannel Modulation for Digital Communications*. PhD thesis, Stanford University, June 1991.

[19] B.R. Saltzberg. "Performance of an Efficient Parallel Data Transmission System". *IEEE Transactions on Communication Technology*, COM-15:805–811, December 1967.

[20] J.T. Aslanis, P.T. Tong, and T.N. Zogakis. "An ADSL Proposal for Selectable Forward Error Correction with Convolutional Interleaving". In *ANSI T1E1.4 Committee Contribution number 92-180*, August 1992.

[21] P.S. Chow. *Bandwidth Optimized Digital Transmission Techniques for Spectrally Shaped Channels with Impulse Noise*. PhD thesis, Stanford University, May 1993.

[22] J.A.C. Bingham and K. Jacobsen. "CATV Reverse Channel Transmission Using Synchronized DMT, Part 1. Overview: Network, Data Rates and Services, Protocol". In *IEEE 802.14 Working Group Contribution number 95-001*, Boston, MA, January 1995.

[23] R. Rom and M. Sidi. *"Multiple Access Protocols: Performance and Analysis"*. Springer-Verlag, New York, 1990.

[24] N. Abramson. "The ALOHA System–Another Alternative for Computer Communications". In *1970 Fall Joint Computer Conference, AFIPS Conference Proceedings, Vol. 37*, pages 281–285, Montvale, NJ, AFIPS Press, 1970.

[25] L.G. Roberts. "ALOHA Packet System With and Without Slots and Capture". *Computer Communications Review*, 5(2):28–42, April 1975.

[26] L. Kleinrock and F.A. Tobagi. "Packet Switching in Radio Channels: Part I–Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics". *IEEE Trans. on Communications*, COM-23(12):1400–1416, December 1975.

[27] F.A. Tobagi and L. Kleinrock. "Packet Switching in Radio Channels: Part II–The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution". *IEEE Trans. on Communications*, COM-23(12):1417–1433, December 1975.

[28] I. Chlamtac and A. Ganz. "Design and Analysis of Very High-Speed Network Architectures". *IEEE Trans. on Communications*, 36(3):252–262, March 1985.

[29] L. Lindner. private communication. July 1993.

[30] R. Lique. private communication. July 1993.

[31] J.A.C. Bingham. Private communication. Amati Communications Corporation, Mountain View, CA, November 1995.

[32] E.R. Bartlett. *Cable Television Technology and Operations: HDTV and NTSC Systems*. McGraw-Hill Publishing Company, New York, NY, 1990.

[33] D. Bertsekas and R. Gallager. *Data Networks.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1987.

[34] J.A.C. Bingham and K. Jacobsen. "CATV Reverse Channel Transmission Using Synchronized DMT, Part 3: Data Access Protocol". In *IEEE 802.14 Working Group Contribution number 95-003*, Boston, MA, January 1995.

[35] A.B. Carlson. *Communication Systems.* McGraw-Hill, third edition, 1986.

[36] R.W. Chang. "High-seed Multichannel Data Transmission with Bandlimited Orthogonal Signals". *Bell System Technical Journal*, 45:1775–1796, December 1966.

[37] M. Dzuban. "Two-way Cable Plant Characterization". In *IEEE 802.14 Working Group contribution number 94-003*, Lake Tahoe, CA, November 1994.

[38] Federal Communications Commission. "Code of Federal Regulations". Title 47, Part 76.605. 1992.

[39] R.G. Gallager. *Information Theory and Reliable Communication.* Wiley, New York, NY, 1968.

[40] B. Harrell. *The Cable Television Technical Handbook.* Artech House, Dedham, MA, 1985.

[41] F.J. Harris. "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform". *Proc. IEEE*, 66(1):51–84, January 1978.

[42] B. Hirosaki, S. Hasegawa, and A. Sabato. "Advanced Groupband Data Modem Using Orthogonally Multiplexed QAM Technique". *IEEE Trans. on Communications*, 34(6):587–592, June 1986.

[43] J. L. Holsinger. "Digital Communication over Fixed Time-Continuous Channels with Memory, with Special Applications to Telephone Channels". *M.I.T. Research Laboratory Electronics Report*, (430), 1964.

[44] K.S. Jacobsen and J.M. Cioffi. "Achievable Throughput of Multicarrier-based Multipoint-to-point Networks Using a Reservation-based Channel Access Protocol". To be presented at Globecom '96, November 1996, London.

[45] K.S. Jacobsen, J.A.C. Bingham, and J.M. Cioffi. "Synchronized DMT for Multipoint-to-point Communications on HFC Networks". In *Globecom '95 Proceedings*, Singapore, November 1995.

[46] K.S. Jacobsen and J.M. Cioffi. "High-performance Multimedia Transmission on the Cable Television Network". In *Proceedings 1994 International Conference on Communications*, New Orleans, LA, May 1994.

[47] K.S. Jacobsen and J.M. Cioffi. "An Efficient Digital Modulation Scheme for Multimedia Transmission on the Cable Television Network". In *Technical Papers, 43rd Annual National Cable Television Association (NCTA) Convention and Exposition*, New Orleans, LA, May 1994.

[48] K.S. Jacobsen, P.S. Chow, and J.M. Cioffi. "Discrete Multitone Modulation for Multimedia Transmission on the Cable Television Network". In *IEEE 802.catv Committee Contribution number 94-024*, Orlando, FL, July 1994.

[49] K.S. Jacobsen and J.M. Cioffi. "Discrete Multitone Modulation for Transmission in the CATV Return Band: Network Issues". In *IEEE 802.14 Working Group Contribution number 94-004*, Lake Tahoe, CA, November 1994.

[50] K.S. Jacobsen, J.A.C. Bingham, and J.M. Cioffi. "CATV Reverse Channel Transmission Using Synchronized DMT, Part 2: Synchronization and Training". In *IEEE 802.14 Working Group Contribution number 95-002*, Boston, MA, January 1995.

[51] A. Karshmer. "Computer Networking on Cable TV Plants". *IEEE Network*, pages 32–40, November 1992.

[52] L. Kleinrock and S.S. Lam. "Packet Switching in a Multiaccess Broadcast Channel: Performance Evaluation". *IEEE Trans. on Communications*, COM-23(4):410–422, April 1975.

[53] E.A. Lee and D. G. Messerschmitt. *Digital Communications*. Kluwer, Boston, 1988.

[54] J.O. Limb. "Performance of Local Area Networks at High Speed". *IEEE Communications Magazine*, 22(8), August 1984.

[55] J.O. Limb and D. Sala. "An Access Protocol to Support Multimedia Traffic Over Hybrid Fiber/Coax Systems". In *Second International Workshop in Community Networking*, pages 35–40, Princeton, NJ, June 1995.

[56] M.A. Marsan and D. Roffinella. "Multichannel Local Area Network Protocols". *IEEE Journal on Selected Areas in Communications*, SAC-1(5):885–897, November 1983.

[57] M. Momona. "A Proposal of a Novel MAC Protocol for CATV Networks". In *IEEE 802.14 Working Group contribution*, Boston, MA, January 1995.

[58] A.V. Oppenheim and R.W. Schafer. *Discrete-Time Signal Processing.* Prentice-Hall, Englewood Cliffs, NJ, 1989.

[59] J.G. Proakis. *Digital Communications - Second Edition.* McGraw-Hill, New York, 1989.

[60] R.S. Prodan, M. Chelehmal, T.H. Williams, and C.M. Chamberlain. "Cable System Transient Impairment Characterization". In *Technical Papers, 43rd Annual National Cable Television Association (NCTA) Convention and Exposition*, New Orleans, LA, May 1994.

[61] A. Ruiz, J.M. Cioffi, and S. Kasturia. "Discrete Multiple Tone Modulation with Coset Coding for the Spectrally Shaped Channel". *IEEE Transactions on Communications*, 40:1012–29, June 1992.

[62] D. Sala and J.O. Limb. "A Protocol for Efficient Transfer of Data Over Hybrid Fiber/Coax Systems". In *Proceedings of INFOCOM*, pages 904–911, San Francisco, CA, March 1996.

[63] C. E. Shannon. "A mathematical theory of communications : Part I". *Bell System Technical Journal*, 27, July 1948.

[64] C. E. Shannon. "A mathematical theory of communications : Part II". *Bell System Technical Journal*, 27, October 1948.

[65] G. Strang. *Linear Algebra and its Applications.* Academic Press, Orlando, FL, 1980.

[66] B. Widrow, J. McCool, and M. Ball. "The Complex LMS Algorithm". *Proceedings IEEE*, 63(4):719–720, April 1975.

[67] B. Widrow and S.D. Stearns. *Adaptive Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1985.

[68] W. Xu and G. Campbell. "A Distributed Queueing Random Access Protocol for a Broadcast Channel". *SIGCOMM*, September 1993.