Adaptive QoS in 802.11e Wireless Networks for Lunar Communications [1,2]

Will Spearman, Jim Martin
Clemson University, School of Computing
Clemson, SC 29632
wspearm@cs.clemson.edu, jim.martin@cs.clemson.edu

Jay L. Gao
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91109
Jay.L.Gao@jpl.nasa.gov

Abstract— Lunar surface communications present specific problems that can be solved with well established terrestrial networking standards such as 802.11. Unfortunately, 802.11 has limitations regarding high priority traffic such as voice and command data which are sensitive to jitter, delay, and loss. The IEEE 802.11e standard provides enhancements that allow traffic with specific needs to be differentiated from normal traffic. While these enhancements have been shown to effectively improve latency and throughput for high priority traffic, they do not offer precise nor consistent control of performance levels. In this paper capacity and priority are explored in the context of lunar communications using 802.11g with 802.11e. We present a method to dynamically optimize 802.11e contention parameters to provide more granular control over the network's quality of service (QoS) for the various flow types. A distributed adaptive algorithm that extends 802.11e's Enhanced Distributed Channel Access (EDCA) is presented. We show that the enhancement provides more granular and consistent performance than that provided by the static algorithm used in standard 802.11e.

TABLE OF CONTENTS

1. INTRODUCTION.....1
2. 802.11 WIRELESS TECHNOLOGY.....2
3. QOS IN 802.11E 4
4. PREVIOUS WORK 5
5. PROBLEM STATEMENT 6
6. METHODOLOGY 7
7. THE DISTRIBUTED ALGORITHM.....8
8. RESULTS AND DISCUSSION.....10
9. CONCLUSIONS AND FUTURE WORK 12
10. ACKNOWLEDGEMENTS 12
11. REFERENCES 12
12. BIOGRAPHY 13

1. INTRODUCTION

In planning for future human exploration activities in a lunar environment, the wireless communications infrastructure used will be of critical importance. Using standardized communication protocols such as 802.11 can reduce costs, enhance interoperability, and provide robust networks that build on proven technology and security.

While long range communications are of great importance, short range communications in a lunar environment are of increased importance as the loading of local communications, e.g., voice conversation and real-time video exchanges among astronauts for situational awareness, is expected to grow exponentially with the number of network elements. The future lunar network will be incrementally realized through a "campaign" consisting of a series of logistical and explorative missions designed to support the infrastructure required for long duration stays and, eventually, for a permanent lunar outpost. A large number of network elements, such as habitat modules, power generation units, mobile robotic vehicles and astronauts, must be networked together through shared access to the network infrastructure. Applications such as Voice over IP (VoIP) will be essential to the day-to-day operation in the lunar outpost. Additionally, deployments must be prepared to support latency sensitive traffic such as command, telemetry, and real-time video. Lunar exploration activities will require remote operation from autonomic control systems or by a team of astronauts. In all possible scenarios high levels of flexibility, service quality, and mobility support must be provided by the network.

There are a number of technical challenges presented by wireless networks due to the shared medium nature of wireless access. Performance problems such as low throughput and high delays in networks with even a small

1

[1] 1-4244-1488-1/08/$25.00 ©2008 IEEE
[2] IEEEAC paper #1152, Version 5, Updated November 17, 2007

number of active stations could jeopardize the mission. As the number of nodes and amount of traffic increases on a wireless LAN, these issues become more serious. While quality of service (QoS) issues have been studied thoroughly in wired LANs, modern high-speed wired networks do not always require complicated QoS due to their relatively high bandwidths, error-free nature, and abundant switching.

Applications that have specific bandwidth or latency requirements might not function well in a congested 802.11 network. VoIP will experience reduced quality during a call when operating over a network experiencing high latencies or low throughput for the connection [8]. Likewise, applications using streaming audio and/or video will experience a reduction in quality. These problems can be more complex than just low throughput and high delay. Large variations in delay can also cause problems with protocols that attempt to adapt to networks with high delay, and in turn cause a considerable degradation in service quality. As wireless technology is adapted to environments with traffic with different QoS requirements, it is important to find solutions for these issues. A flexible IP-based, self-organizing, multiple access network will enhance the operability of future missions that are complex and involve large numbers of mobile nodes, but at the same time a new approach to bandwidth management is needed to provide isochronous services to real-time critical data flows over a background of dynamically fluctuating non-real-time telemetry and file transfers.

The IEEE 802.11e standard provides enhancements that allow traffic with specific needs to be differentiated from best-effort traffic. While these enhancements have been shown to effectively improve latency and throughput for high priority traffic, they do not offer precise control of performance levels. For example, strict prioritization might starve low priority traffic. While this could be viewed as acceptable considering the traffic isn't viewed as important, in many situations network conditions could allow both high and low priority traffic to find a balance where both experience acceptable service while preserving their priority. Further complication is caused by the interaction between the 802.11 MAC layer, higher layer protocols, and applications requiring different service levels. The optimal 802.11e MAC QoS settings change as network dynamics evolve over time.

We present a method to dynamically optimize 802.11e contention parameters to provide more granular control over the network's quality of service (QoS). A distributed adaptive algorithm that extends 802.11e's EDCA is presented. We show that the enhancement provides more granular and consistent performance than that provided by the static algorithm used in standard 802.11e.

This paper is organized as follows. We provide a brief background on 802.11 including QoS enhancements provided by 802.11e. Next we summarize related work. We describe our methodology and testing process. Then we describe preliminary work carried out to identify and illustrate the problem statement. Finally we describe the distributed algorithm and show simulation results that demonstrate the benefits of our method. We conclude by identifying future work and ways to further test the algorithm presented.

2. 802.11 WIRELESS TECHNOLOGY

The 802.11 set of standards covers telecommunications and information exchange between systems in local and metropolitan area networks [1]. The 802.11 standard provides best-effort packet services for the Medium Access Control (MAC) layer of wireless networks. This MAC layer provides wireless stations with fair access to the medium in a best-effort manner. In the following sub-sections we review the base 802.11 MAC protocol.

The 802.11 MAC Layer

The original 802.11 MAC layer is built around two coordination functions that control medium access by the use of distributed coordination and centralized coordination. In the Distributed Coordination Function (DCF), the access control mechanisms are located at the station as opposed to the Point Coordination Function (PCF) in which control is centralized to the Access Point (AP). In 802.11 networks, DCF is always used, although PCF may be used optionally along-side DCF. Most 802.11 products support just DCF.

Collision Management in 802.11

In 802.3 Ethernet networks, the primary method for medium access is Carrier Sense Multiple Access with Collision Detection (CSMA/CD) in which collisions are detected on the channel, and are handled by back-off counters that reduce future collisions by randomly increasing window sizes. Detection and recovery are efficient and feasible in wired networks due to the high bandwidth and low packet times of modern networks. This "reliable" nature of wired networks significantly reduces the impacts of collisions. In the wireless realm, however, interference can cause substantial noise resulting in frequently corrupted packets. Even more problematic is the fact that channel sensing is not possible since most radios can not simultaneously send and receive [1]. For these reasons collision detection is not possible for 802.11 wireless networks motivating the need for CSMA with Collision Avoidance (CSMA/CA).

CSMA/CA works on the principle of listening before transmitting. By using wait times efficiently, this can lead to an algorithm by which all stations are allowed to gain access to the medium in a relatively fair manner and minimizing collisions using DCF or PCF, or both. This algorithm relies on inter-frame spacing to coordinate the

communication of the stations. Four specific time intervals are defined as follows:

- SIFS (Shortest Inter-Frame Space): the wait time between the last transmission and high priority transmissions such as Request To Send (RTS) and Clear To Send (CTS) frames and positive acknowledgments (ACKs). Positive ACK frames are given priority so that currently communicating stations are given immediate feedback on the most recently sent frame. Since RTS and CTS frames are control frames, they are naturally given priority over other frame types.

- PIFS (PCF Inter-Frame Space): the minimum idle time for contention-free access such as PCF; this is discussed in detail below. This interval allows the point coordinator priority over stations.

- DIFS (DCF Inter-Frame Space): the minimum idle time for contention based access such as DCF. Any station may claim the medium after this time interval.

- EIFS (Extended Inter-Frame Space): minimum wait time for a station that receives corrupted frames.

The frame spacing intervals allow DCF and PCF to interact seamlessly and with as few collisions as possible by always assuming the following relationship: slot time < SIFS < PIFS < DIFS < EIFS. The follow diagram shows the relationship of the inter-frame spaces to the access algorithms discussed in the next sections.

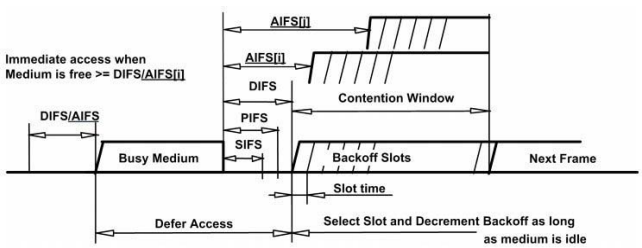

Figure 1. 802.11 transmission intervals

In Figure 1 taken from [2], it is clear that components that are controlled by smaller time delay intervals will have a distinct advantage over those that use longer time intervals. In the following sections, it will be shown how these inter-frame spaces are the basis for providing control mechanisms with priority over stations, as well as providing station priority in the absence of centralized control.

DCF

DCF allows stations to transmit without a central coordinator. When a station wishes to transmit, and has sensed that the medium is free, it waits for a DIFS and transmits. If during the DIFS, the medium becomes busy, it begins decrementing a back-off counter that is defined by the Contention Window (CW). The CW begins equal to CWmin and ends equal to CWmax. After each consecutive collision, the counter is set to a random value between 0 and CW. Each time a collision occurs the CW is increased until it equals CWmax.

If the CW reaches zero, and the medium is still free, then the station begins transmitting. If during the countdown, the medium is seized by another station, the station stops the counter and resumes after the transmission period. If the station senses the medium to be free, reaches a counter value of zero, and begins a transmission that results in a collision (no ACK received), the station will pick a new CW value.

DCF also includes an optional RTS/CTS mechanism to eliminate the hidden station problem. The hidden station problem occurs when two stations can sense transmissions of the AP, but not of each other. Due to their inability to receive each other's signals, the two stations can claim the medium simultaneously, and will cause a collision at a central destination. To prevent this, before sending a frame, a station transmits a RTS packet and then receives a CTS packet from the central station. Both of these frames include information regarding the time it will take to send the frames, which allows other stations to set a timer called the Network Allocation Vector (NAV) since the medium will be busy at least for that length of time. After that time, stations begin normal time interval waiting, and back-off counter decrementing. Since RTS and CTS frames are allowed to be transmitted after a SIFS, they have priority over normal DCF transmissions.

PCF

In PCF, medium access is controlled by a Point Coordinator (PC). The PC controls access by looking for stations wishing to transmit during a Contention Period (CP), and polling stations during a Contention Free Period (CFP). Together the CP and the CFP form a superframe which repeats for each time period. During the CFP, PCF is used to control access, and then during the CP, DCF is used.

The CFP portion of the superframe begins with a beacon frame that contains management information such as protocol parameters and time synchronization. After the beacon frame has been transmitted, the PC polls stations in a round-robin manner, and upon successful response, allows the station to transmit either an ACK indicating it has nothing to send, or a DATA+ACK frame. Having received no response from a station the PC moves on, and the station is not allowed to transmit until the CP, or during the next CFP. The CFP ends when the time period specified by the beacon frame expires, or a CFP-EndFrame is sent. After the CFP has ended, a normal DCF period proceeds. However,

since PIFS is shorter than DIFS, the PC can immediately seize the medium and begin another CFP if desired.

While PCF was intended to provide a form of QoS to 802.11 networks, it is generally agreed that it fails to provide this service adequately. Although PCF gets priority over DCF since the PIFS is always less than the DIFS, it suffers from the fact that individual network flows cannot be singled out for prioritization since the PC polls in a round-robin fashion. High priority can be given to individual stations, but affecting service on a more granular level is impossible with PCF. Also, polling can result in excessive overhead and large end-to-end delay when the number of stations is large [8].

3. QoS IN 802.11E

In an effort to give 802.11 networks true QoS, 802.11e was standardized. 802.11e introduced enhancements to the existing DCF and PCF, placed them under the heading of the Hybrid Coordination Function (HCF). The HCF is comprised of Enhanced Distributed Channel Access (EDCA), which is an enhanced DCF, and HCF Controlled Channel Access (HCCA), which has many traits in common with PCF. These two access methods work separately or together, just as in 802.11, where DCF is mandatory and PCF is optional. While the fundamentals of the original functions were not changed, augmented information allows HCF to provide QoS to specific flows and/or stations.

EDCA

In the EDCA, MAC layer parameters are used to provide priority to each traffic class (TC) in a contention access manner similar to DCF. The parameters that can be manipulated are the Arbitration Inter-Frame Space (AIFS), the Transmission Opportunity (TxOp), the CWmin, and the CWmax. These parameters are given default values at each station for each TC, or they can be overridden by an AP using special coordination frames.

Each parameter creates priority in a different way. The AIFS focuses on the time interval the TC must wait before trying to gain the TxOp. The TxOpmax defines the length of time a station may transmit on behalf of a TC. The CW parameters prioritize by adjusting the back-off counter's minimum and maximum sizes respectively. Each parameter can be varied within TCs, while normal DCF rules apply between TCs within a station. For example, each TC is maintained in a separate queue within a station, and each queue contends for access to transmit on behalf of the station. If the back-off timers of two TCs within a station expire at the same time, the algorithm treats this as a virtual collision and will requires the lower priority flow to increment its collision counter and find a new back-off counter value.

In EDCA, the AIFS is a time interval that is equal to or greater than the DIFS such that higher priority stations can be given low values. When the AIFS expires, normal DCF operation for a station continues by decrementing the back-off timer. Therefore, TCs with low AIFS values will be more likely to gain access to the medium. Similarly, the CWmin value controls the size at which each TC starts its CW. A lower CWmin allows the flow to seek the medium sooner after the AIFS, and to be more likely to get the medium after a collision. When a collision does occur, each TC multiplies the current CW by the Persistence Factor (PF, usually equal to 2) to calculate the next back-off timer. The CWmax controls the maximum value to which a flow's CW can grow. A larger value will allow the flow to be less competitive during heavy collision, high load situations. Lower priority flows will have larger CW values and will wait longer when network traffic is causing many collisions. Finally, the TxOpmax controls the length of time for which a station can transmit on behalf of each TC. Larger TxOpmax values allow stations to send more frames during each use of the medium.

HCCA

In the HCCA portion of the HCF, a Hybrid Coordinator (HC) is used to centrally manage the medium in much the same way as PCF, with the exception that HCF uses parameterized QoS. Parameterized QoS refers to the use of specific information such as minimum data rate, maximum latency, etc. that allows the HC to prioritize accordingly. When acting on behalf of a new TC, the station sends the HC the requirements of the TC. The stations transmit these requirements in the form of Traffic Specification (TSPEC) frames to the HC, which can accept or reject the request based on network conditions. Unlike PCF, the HC can poll stations during the CP, which allows the HC to provide TC prioritization as well as station prioritization.

Similar to the relationship between DCF and PCF, EDCA and HCCA are integrated within 802.11e yet operate independently. Even when HCCA is not used, EDCA uses superframes with CFPs and CPs. If a HC does not exist, or chooses not to poll during a superframe, the CP starts and EDCA rules are followed. The HC's transmission is governed by the PIFS time interval, and since this interval is smaller than the AIFS, the HC can obtain the medium before normal stations do. If the HC is active, it polls stations that have indicated a need to be scheduled, allows them to transmit, then ends the CFP with a CF-End control frame.

Other 802.11e Enhancements

The 802.11e standard defines the HCF in order to provide specific flows and stations with high priority over others, but also defines other mechanisms to indirectly aid in this goal, such as: Contention Free Bursts (CFB), Special ACKs, and Direct Link Protocol (DLP).

Under normal operation, a station must contend for channel access after each TxOp. This can result in low throughput for data-rate sensitive applications even in the situation where it is given priority over other stations. Using CFB, if there is still time left in the TxOp after a frame is transmitted the station is allowed to continue without waiting for a SIFS or the back-off counter. By using CBF in conjunction with higher TxOpmax values, CFB can reduce the amount of overhead and allow a high priority station that already holds the medium to achieve greater throughput.

The second indirect improvement is the addition of two options to the QoS control field of data frames regarding ACKs. A station has the option to send packets with a flag set such that an ACK for that packet is not generated. In order to increase efficiency the No Acknowledgement (NOACK) flag can be used for applications where ACKs are not important, or do not signify any action on the part of the sender. Streaming media, for example, can tolerate packet loss but suffers greatly in the event of high latency. NOACKs in this situation improve efficiency by eliminating unnecessary acknowledgement for the receiver since no retransmission attempt will be made for real-time data. A second type of ACK called the BlockACK is also defined as aggregated ACKs that accumulate during a CFB. An ImmediateBlockACK can be requested by the sender after a CFB, and if necessary, the receiver can respond with a DelayedBlockACK if the receiver cannot respond before the sender's request timeout. These ACKs allow CFBs to be used effectively in an 802.11e environment, but do not directly improve prioritization. Block acknowledgement has been a particularly useful feature in space communications characterized by asymmetry and capacity constraints and is utilized in several protocols standardized by the Consultative Committee on Space Data and Standards (CCSDS). The CCSDS File Delivery Protocol (CFDP), which is the space-version of the File Transfer Protocol (FTP), uses block acknowledgement to save bandwidth over the long-haul uplinks that are typically one order of magnitude lower in capacity than the downlinks. While such asymmetry is not as significant for surface networking, it is expected that the ability to control acknowledgement overhead for individual TCs will be important for capacity planning for future missions.

Basic 802.11 operations allow ad-hoc networking directly between stations or infrastructure-based networking where stations cannot communicate directly. However in 802.11e DLP allows networks to use both simultaneously. When a station wishes to use DLP, it sends a DLP request to the AP. The AP forwards the request to the receiver. If the receiver supports DLP it will send a response through the AP back to the sender. The sender will then contact the receiver directly and begin the transmission. This direct communication can be especially useful when two stations are located near each other, but distantly from the AP. The signal between the two stations may be stronger and could result in fewer dropped frames. More importantly, eliminating the extra step of going through the AP can reduce the round trip times by half and lower the load on the AP for forwarding unicast traffic. The DLP feature could be very desirable for supporting local voice loops and potential video streaming between two 802.11 stations.

Current Status of 802.11e

The lack of clear market requirements for 802.11e and additional product complexity has resulted in few vendors shipping a full 802.11e implementation [3]. In an effort to stimulate development towards full 802.11e deployment, the Wi-Fi Alliance (WFA) developed requirements for hardware to be Wi-Fi Multimedia (WMM) compliant and Wi-Fi Scheduled Multimedia (WSM) compliant [3]. These two standards are subsets of 802.11e that allow stations to support applications that would benefit from prioritization. WMM uses EDCA while WSM uses HCF, but neither includes the other enhancements discussed in the previous sections. WMM uses four categories in which to place traffic: voice, video, best effort, background. Each of the eight 802.11e categories are mapped evenly to a traffic class to allow backwards support for non-WMM stations. Any station sending data that is not assigned a traffic class, is considered best effort traffic. In this paper we refer to an 802.11b node (or just a ‘b’ node) as an 802.11e node configured to use the standard best effort traffic class. We refer to an 802.11e node (or just an ‘e’ node) as an 802.11e node that has at least one of its contention parameters configured to give the station priority over a best effort node.

4. PREVIOUS WORK

Previous work in the area of 802.11 QoS has shown that 802.11 network parameters can be adapted to provide better overall network service to general clients by maximizing throughput based on current network conditions [6, 7]. It has also been shown that existing 802.11 designs allows parameter tuning such that services like VoIP can be accommodated and given some QoS guarantees [8]. The fact that these approaches do not accommodate multiple priority levels or standardized parameters has lead to the development of 802.11e. These developments, and their relevance as the basis of an adaptive algorithm, are discussed in the remainder of this section.

In [10], the importance of dynamic solutions to service prioritization becomes clear in the results presented. It is shown that the design of 802.11e increases the possibility of collisions and increases delay by adding an extra contention layer. In normal 802.11 networks the only collisions that can occur are those between stations as they try to gain access to the medium to transmit the packet at the head of the single send queue. In 802.11e, virtual collisions can occur when each TC queue must contend for access within

the station's queue manager, as well as real collisions when the winning queue is allowed to transmit. The results in [10] show that throughput in 802.11e is decreased and latency is increased when compared to 802.11a networks due to this extra contention being introduced. Therefore, dynamic and aggressive tuning of the network parameters is required to achieve benefit from prioritization.

In [6, 7] the authors evaluate a mechanism which allows dynamic tuning of the timing used in the back-off algorithm in 802.11. They showed that a dynamic algorithm based on the number of currently active stations, which manipulates the minimum back-off time can allow a wireless network to perform closer to the theoretical capacity of the medium. The algorithm estimates the current performance and adjusts its back-off accordingly. Their findings show that static network parameters lead to under-utilization of the medium and show the importance of a dynamic algorithm. However, the network parameters are adjusted with global knowledge of the number of nodes in the network and collision rates.

In [9] the authors investigate two methods of choosing CWmin in 802.11 networks based on proportional fairness and time-based fairness. They conclude that proportional fairness in a network based on weights provides higher throughput than time based fairness. Their work shows that CWmin can be tailored to a network to provide all nodes with fair access to the medium if priority mechanisms are used, and therefore it follows that the same principles can be used to provide nodes with unfair access, or differentiation, using contention window parameters.

In [4], the authors evaluate how a network with 802.11b and 802.11e nodes performs with different EDCA contention parameters, and how the delay and throughput are affected by these parameters. The 802.11b nodes model background traffic while the 802.11e nodes model high priority traffic. Four different contention parameters are tested: the initial size of the contention window, CWmin, the maximum size of the contention window, CWmax, the Arbitration Inter-Frame Spacing interval (AIFS), Persistence Factor (PF). These parameters can be adjusted to differentiate 802.11e traffic from 802.11b traffic present on the same network.

In [4], AIFS was shown to be the most effective contention parameter for protecting high priority traffic from background traffic. However, the authors show that using PF and CWmin for differentiation may have the advantage of allowing for better performance of the low priority traffic. The CWmin parameter can be characterized as a compromise between AIFS, which is the most effective for high priority nodes, and no differentiation at all, which is the least adverse towards low priority nodes.

The work presented in [11] is similar to our research. The authors use a two-level approach to fair, yet prioritized service. The first level of protection for high priority services guarantees that changing network conditions do not affect data streams such as VoIP and video that have constant QoS requirements. By using budgeted TxOp values for each queue, new flows are not allowed to have immediate access to their share of bandwidth regardless of their TC or parameter settings. This ensures that established flows are not disrupted by new flows. The second level of protection, called Fast-Backoff with Dynamic Adjustment when Fail or Successful, is the most similar to our work due to its distributed nature. Under this scheme, when a station experiences a transmission failure, its CW is increased by a factor greater than 2 which results in a faster than exponential backoff. In addition to the CW increase, the station's CWmin is increased by a factor. When the station experiences a transmission success, CWmin is decreased by a factor, and the CW is reset to CWmin. This dynamic adjustment results in a dramatic decrease in the number of collisions, as well as more reliable service for the voice and video data. This method adjusts the EDCA parameters on all successes and all failures with no regard for direct network performance measurements. The approach in [11] differs from the approach presented here in that the Adaptive Algorithm uses direct network performance measurements as opposed to using transmission successes and failures. Also, [11] uses the default 802.11e TCs under normal circumstances, and does not allow fine grained performance control.

Finally, as shown in [9], 802.11e parameters can be tuned based on network conditions to allow better performance than a single setting. Although these settings are not changed dynamically in this study, they do show that changing network conditions require changing parameters to use the channel efficiently.

In works such as [13], the HCCA is shown to be better at efficiently using the medium than pure EDCA. Contention in HCCA networks is reduced since the central coordinator controls access in an organized manner. Although the stations still use EDCA at the station level queues and during CPs, the central coordinator controls most of the transmission opportunities. Despite this result, EDCA can still be considered to be an important research topic from a reliability perspective. In the HCCA scheme, the AP is a single point of failure. If the AP fails, the network will fall back on Ad-hoc mode with EDCA controlling access.

5. PROBLEM STATEMENT

The first set of preliminary experiments in [12] was designed to replicate the data in [4] which showed the effect each 802.11e contention parameter has on performance when adjusted individually. As found in [4], the results suggest that prioritization based on AIFS provides the best level of service for high priority traffic since in all cases it provides the lowest access times. Prioritization based on CWmax provides the next best service and prioritization

based on CWmin provides the least aggressive service. These finding serve as the basis for parameter adjustments that are discussed in the following section.

Also presented in [12] is a second set of baseline experiments that were performed to further motivate the need for tighter QoS controls in 802.11e. The primary contribution of these experiments was to show that an adaptive algorithm can provide a continuum of performance levels between the static categories of TC0 and TC3, and that conversely, the static categories do not allow arbitrary choices of network performance. It was shown that an appropriate choice of parameters could replicate the performance of the static categories as well as the performance levels in between. The experiments that were shown used a simplistic and static network model.

Problem Statement

In prior work we have shown that:

- It is difficult to know the optimal value for the 802.11e contention parameters.
- The effective performance relative to different traffic classes depends on network dynamics.
- Adjusting priority dynamically is possible for simple networks; however, more complicated networks may present greater difficulty.

The work presented here furthers the work presented in [12] by using a more realistic network composed of combinations of traffic and transport types intended for actual deployment in a lunar communications system.

The algorithm presented in this paper attempts to solve these problems by adjusting the EDCA parameters independently at each station based on a single value assigned to each data stream. This single value determines the threshold for when a stream begins to adjust its parameters, or in other words, how aggressive the stream is in claiming access to the wireless channel. The following sections explain the methods used to accomplish this goal.

6. METHODOLOGY

Figure 4 shows the simulated network model used in the research reported in this paper. The ns2 simulator version 2.28 was used with an EDCA add-on [5] to simulate an 802.11e network. The model represents a possible scenario for lunar communications. The Lunar Communications Module (LCM) is an access point for the 802.11g nodes to an earth-bound network through a high speed Ka Band isochronous connection. For these simulations the speed, latency, and response characteristics of the connection between the LCM and the Earth is not considered in order to focus on the local communications among the 802.11g nodes. The LCM is assumed to perform an appropriate amount of buffering and error control for this type of connection. Therefore, this portion of the network is only shown to demonstrate the scenario context. Furthermore, end-to-end connections are made only between wireless stations on the lunar surface and the LCM, with the LCM acting as a data warehouse and relay station. All performance metrics are assumed to be between wireless stations and the LCM.

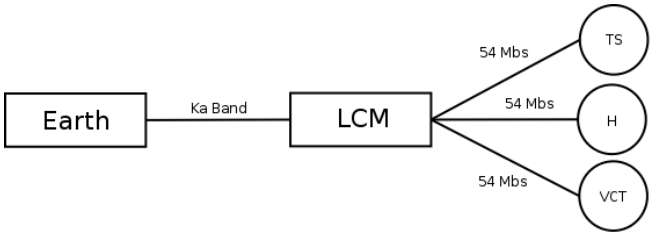

Figure 4. Simulation network structure

The simulated lunar surface network is composed of unmanned science stations and manned vehicles. The unmanned science stations transmit telemetry and standard definition (SD) video (represented by “TS” in the figure). The stations might also occasionally transmit high definition video (represented by “H” in the figure). Examples of manned vehicles are humans in Extra Vehicular Activity (EVA) suits, and humans in powered vehicles. Both of these human components transmit voice, command, and telemetry data (represented by “VCT” in the figure). Table 1 shows the Traffic Class values used in the simulations. TC0 represents the highest priority, while TC3 represents the lowest priority. TC2 is considered the default values for traffic that is unclassified.

Table 2 shows the traffic characteristics of the data flows used in the simulations. The workloads are based on reasonable voice, video, and data profiles that might be used in a lunar colony. The profile of G.729 VoIP traffic is modeled with a constant bit rate (CBR) traffic generator configured with a sending rate of 8 Kbps. In addition to VoIP, the VCT generates low bandwidth, but high priority command data. Examples of command data include instrument commands, EVA suit health status, or other such mission critical operational data. All command data generated by a VCT is modeled as a CBR traffic generator transmitting at a data rate of 13 Kbps.

SD and HD video are used at manned and unmanned stations. Since not all visual applications require the higher resolution of HD video, SD video is the default output for unmanned monitoring stations. SD video is generated from all unmanned stations using variable bit rate (VBR) data at 4 Mbps. HD video is only generated when higher resolution

is needed for a specific application at a manned station. Therefore HD video is generated only at a few nodes per simulation.

Table 1. 802.11e Traffic Classes

TC / Param	TC0	TC1	TC2	TC3
AIFS	2	2	3	7
CWmin	7	15	31	31
CWmax	15	31	1023	1023

SD and HD video profiles used were taken from the H.264/AVC standard [15]. Both profiles used in this study are not intended to specifically represent an exact combination of resolution and dimension, but instead could represent the following profiles: SD video at 4 Mbps could represent Extended Profile Level 2.1 while HD video could represent High Profile Level 3. These profiles can describe any number of combinations of resolution and frame rate, and are only approximations of video traffic that could be used in these scenarios.

The video traffic model incorporates and on and an off time with durations determined by a Pareto distribution. During an on state, the model transmits a burst of data at a configured maximum burst rate. The Pareto model's mean and shape parameter were set to 1.4, an "on" time of 5 seconds, and an "off" time of 1 second.

Table 2. Traffic Types and Attributes

Type / Attribute	Voice	Command	Telemetry	SD Video	HD Video
Protocol	UDP	TCP	UDP	UDP	UDP
Profile	CBR	CBR	CBR	VBR	VBR
Rate (kbps)	8	13	248	4000	12000
Packet (bytes)	20	1000	1000	2048	2048
AR_PRESET	1.5	5.0	15	35	35
Normal Priority	TC0	TC1	TC2	TC3	TC3
Metrics	Throughput, Jitter	Round Trip Time	Throughput, Jitter	Throughput, Jitter	Throughput, Jitter

For these simulations the EDCA code was extended to allow data collection at the transport layer. The main metric used to evaluate the performance of the system with the Adaptive Algorithm is the MAC access delay, which is the time from a packet's entrance to the MAC queue until the time it is successfully sent. Arriving packets were time stamped at queue entry and at packet transmission so that the average delays could be calculated. Using the access delays, a mean access delay is accumulated in between algorithm actions. Using the mean access delay as a measure of algorithm performance is discussed further in [12].

In addition to mean access delay which is used in the algorithm, we also obtain a number of application specific performance measures in order to evaluate the network. These measures include end-to-end throughput, jitter, Mean Opinion Score (MOS), and Round Trip Time (RTT). The metric used for each type of data is shown in Table 2.

7. THE DISTRIBUTED ALGORITHM

The adaptive algorithm tunes the AIFS, CWmin, and CWmax parameters based on the performance of the network that is measured locally using a measure referred to as the aggressiveness ratio (*ar*). The *ar* is the ratio of the average access delay to an estimate of the minimal access delay. In other words, the *ar* is a measure of how aggressively the data flow should adjust its parameters. An *ar* for each flow is calculated by measuring its current performance and comparing it to the best possible performance it can hope to achieve. The *ar* value specifies a relative performance measure of how the station is performing compared to how well it could perform if there were no other channel contention.

From this point forward, AR_PRESET refers to the target *ar* value that is set before the simulation for a data flow, and *ar* refers to the aggressiveness ratio calculated for each data flow during the algorithm operation. The AR_PRESET is the overall performance goal of the data flow, while the *ar* is the measure of the progress toward that goal at a particular point in time. Figure 7 shows the pseudo-code that corresponds to the algorithm discussed in the rest of the section.

By dynamically adjusting contention parameters while the network is operating, the high priority nodes are able to maintain their quality of service under various levels of network load. This dynamic quality is important since it allows configurations for any delay requirements, rather than the narrow performance characteristics of TC0. The algorithm presented in Figure 7 uses a single setting called the AR_PRESET that can predictably provide performance on the continuum of access delays between TC0 and TC2.

One of the primary concerns for the algorithm is providing a way for each node to derive stateful information about the system based on local observation rather than global knowledge. In the context of space communication, global knowledge is, to some extend, easier to obtain because the communication scenarios are usually planned out in detail. Typically one knows at any given time the number of active stations and the expected traffic load from each, therefore optimizing the back-off algorithm based on such information could be implemented without undue difficulty. However, lowering dependency on global knowledge is still valuable for the future of NASA missions not because such information is difficult to obtain, but instead we want the

network the adapt to changing mission scenario automatically, therefore reducing the operational complexity and cost associated for network configuration and increase the robustness of the network during contingency situations.

During operation the algorithm uses mean access delay samples from before and after changes to the QoS values in order to measure success or failure of the parameter adjustment. After the change in access delay is assessed, the delay is saved, and a new time period begins. This feature allows the station to adapt quickly to current conditions

```
FOR random in 0 to STARTUP
    do nothing
ENDFOR
aifs = AIFS_OPTIMAL
cwmin = CWMIN_OPTIMAL
cwmax = CWMAX_OPTIMAL
FOR counter = 0 to START_VAL by 1
   optimal_delay += delay
ENDFOR
optimal_delay /= counter
aifs = AIFS_TC2
cwmin = CWMIN_TC2
cwmax = CWMAX_TC2

normal_operation:

interval_delay += current_delay
IF (time_out < now)
   interval_delay /= INTERVAL
   ar = interval_delay / optimal_delay
   ar_ratio = ar / AR_PRESET
   IF (ar_ratio in range(GOODL, GOODH)
      break
   ELSEIF (ar < AR_PRESET)
      IF (ar_ratio < LOW)
         aggressive_decrease()
      ELSE
         relaxed_decrease()
      ENDIF
   ELSEIF (ar > AR_PRESET)
      IF (ar_ratio > HIGH)
         aggressive_increase()
      ELSE
         relaxed_increase()
      ENDIF
   ENDIF
   time_out = now + INTERVAL
   interval_delay = 0
ENDIF

GOTO normal_operation
```

Figure 7. Adaptive Algorithm Pseudo-code

The second important design decision involves finding a reliable method of providing a node with knowledge of its overall goal for its access time, and more importantly, knowledge of when to give up pursuit of that goal. Since the only value known to the algorithm at start-up is the AR_PRESET, which describes how aggressively a data flow treats large *ar* values, the node needs a way to discover the network's current performance status. As a solution to this, a minimal mean access delay is measured when the node comes online. During Adaptive Algorithm start-up, a node is given priority over all other nodes in order to estimate the minimum access delay to the base station. For this short period, the node lowers all parameters and assumes highest priority to get a "best case" measurement, to which it will then compare its mean access delays gathered later. This measurement is not intended to measure any physical or link layer capabilities, but rather to find the performance possible if the medium were not being shared with other nodes, or in other words, if there was no contention.

After the start-up phase has completed, the node's normal operation begins. A node's normal operation consists of increase and decrease adjustments to the EDCA values. The goal of the adjustments is to keep the *ar* in an acceptable range. Adjustments are made based on comparing the current *ar* value to the AR_PRESET for the station. This derived value is called the *ar_ratio*. Table 3 shows example values that demonstrate how the *ar_ratio* of the current *ar* to the AR_PRESET influences parameter changes. An "a" denotes an aggressive change while an "r" denotes a relaxed change. A "-" denotes a decrease in priority while a "+" denotes an increase in priority. The distinction between an aggressive change and a relaxed change is made by the order in which the parameters are changed. For example, according to the ordering explored in [4], an aggressive increase in priority would first try adjusting AIFS until exhausted, then try CWmax, then if the other two could not be adjusted it would try CWmin. The upper and lower bounds of adjustment are the static TC0 and TC3 parameters. A parameter is considered exhausted when it has been increased in priority to TC0 or decreased in priority to TC3.

Even though only small adjustments are made when a data flow is near its AR_PRESET goal, volatile network conditions can result since the *ar* will never be equal to the AR_PRESET. In other words, the algorithm by itself can never reach a stable condition and will fluctuate within the window of small adjustments. For this reason, a tolerance mechanism was added. The tolerance value adds padding to the decision to make parameter adjustments. When the *ar* has been in the acceptable range for a period of time and the tolerance is high, the *ar* must be out of the acceptable range multiple times to trigger a change. Conversely, when tolerance is low, the *ar* must be in acceptable range multiple times in order to guarantee that an adjustment will not be made the next time an unacceptable *ar* is found.

Table 3. Example Parameter changes

ar Ratio	< 0.5	0.5 – 0.8	0.8 – 1.2	1.2 – 2.0	> 2.0
Change	a-	r-	none	r+	a+

8. RESULTS AND DISCUSSION

Figures 8 through 10 show the data collected using the network scenario described in Section 5. For each graph ten data points were measured that consist of three types of nodes: Voice/Command/Telemetry (VCT), Telemetry/SD Video (TS), HD Video (H). The data point number designates the number of VCT nodes and TS nodes while the number of H nodes is fixed at two for all points. For example, data point 4 has four VCT, four TS, and two H nodes for a total of ten nodes.

Figures 8 through 10 show data compared across three simulations types: simulations using no prioritization (NoDiff), using normal static prioritization (Normal), and using the Adaptive Algorithm (AA). In the NoDiff scenario, no differentiation is used by disabling the Adaptive Algorithm and setting all priority queues to the default TC2 values shown in Table 1. According to the 802.11e standard, data flows coming into the AP that do not have TC designations are treated as TC2.

In simulations using Normal prioritization, each data flow is set to an appropriate TC as shown in Table 2. Voice and Command are considered most important since they represent basic communication between mission critical components. Voice is the direct line of communication between EVA suits, and Command data is the direct line of communication between automated systems. While in many cases telemetry information can be considered mission critical, here, telemetry data is treated as supplemental information and is given the next highest priority. Finally, video has the lowest priority since it is considered an alternate form of communication to Voice and Command data. There exist scenarios where each of these data flows can be mission critical, and using the Adaptive Algorithm, the priorities could be adjusted based on the task at hand; however, for these simulations it was determined that the priorities were as specified above.

In Figure 8 the average throughput of each node was totaled for each type of prioritization into a total throughput for the network. This figure highlights an advantage of the Adaptive Algorithm. The fact that the algorithm does not engage aggressively unless the AR_PRESET requirement is not being met allows many situations where the data flows can coexist without the need of increased prioritization. In fact, the algorithm will reduce parameters to TC3 settings if the current *ar* is too low for its AR_PRESET. This feature allows better than NoDiff performance when contention in the network is low. In the first data point for AA that contains one VCT, one TS, and two H nodes, the Voice and Command data flows do not add additional contention to the network as they would under the Normal scenario. Since Voice and Command use low priority parameters, the low priority video flows are not impacted as greatly as they are under Normal prioritization. For this same reason NoDiff performs better than Normal, and AA performs better than both for the first two data points. As the network load increases Normal performs better than AA and NoDiff.

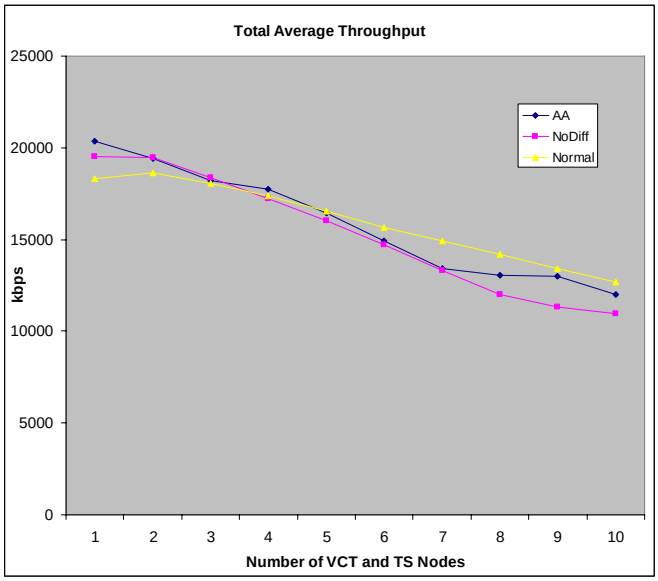

Figure 8. Total Average Throughput

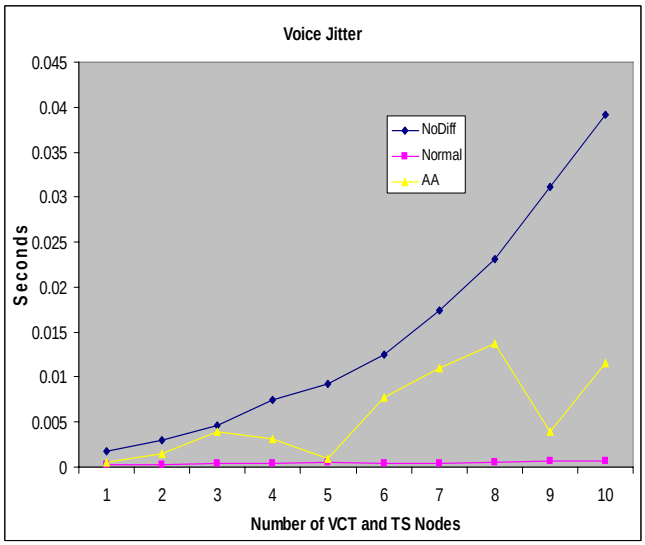

Figure 9. Voice Jitter

Figure 9 shows the average jitter for all voice flows for each of the three prioritization methods. This data points out the disadvantage of dynamic prioritization. By starting out with default values, each data flow must find its own balance of parameters that satisfies its AR_PRESET. In doing so, network conditions can become more volatile than Normal prioritization. This can put more strain on network components such as codec buffers, but as the graph shows, the AR_PRESET can be chosen so that the jitter still stays under an acceptable level. Compared to NoDiff, AA still succeeds in delivering lower jitter, but it cannot be as effective as Normal prioritization.

Figure 9 also highlights the difficulty in designing a dynamic algorithm that is stable for most network configurations. For data point 5 and 9 jitter is difficult to predict based on the AR_PRESET. This variability is most likely a result of the tolerance mechanism presenting artifacts during certain network configurations. If the algorithm chooses too aggressively for too many data flows, the contention parameters are set more aggressively, and performance in the network suffers. Although jitter remains better than NoDiff for all data points in Figure 9, providing more consistent behavior should be a primary goal for future versions of the AA.

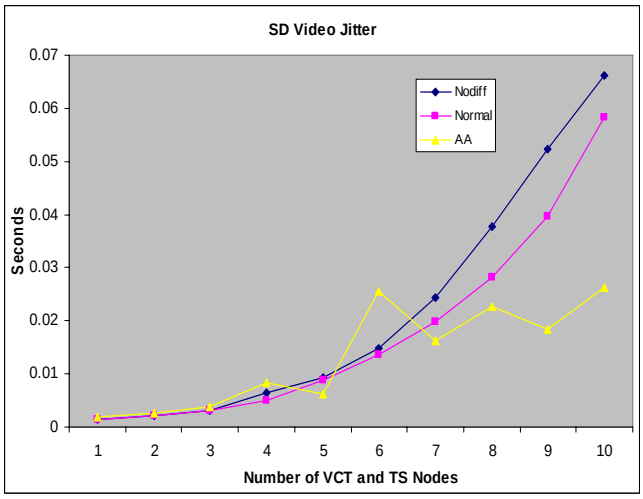

Figure 10. SD Video Jitter

The average jitter for SD video is shown in Figure 10. This data represents a lower priority level, and as compared to voice jitter in Figure 9, is able to provide better predictability. The data still shows that performance for AA experiences greater variability than Normal, but over most data points is better than both NoDiff and Normal. HD video performance is similar to SD but has greater overall jitter, and is therefore not shown.

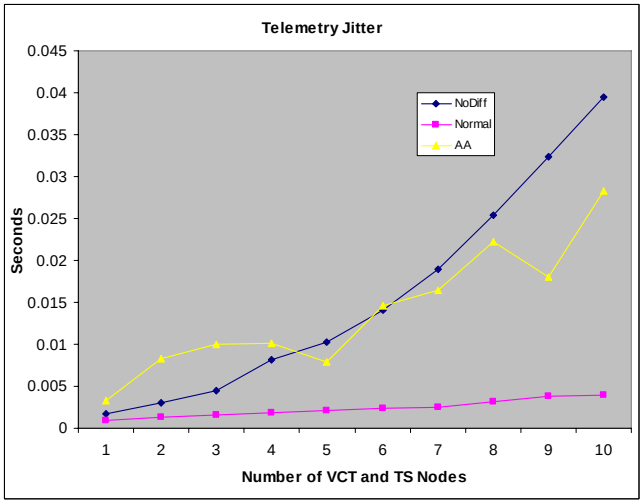

Figure 11. Telemetry Jitter

In Figure 11, which shows the jitter for Telemetry, the data exhibits similar characteristics as the data for voice jitter. Due to inconsistencies in the settling of the algorithm, performance does not follow the same curve as NoDiff or Normal, however, for most data points the performance is better than NoDiff.

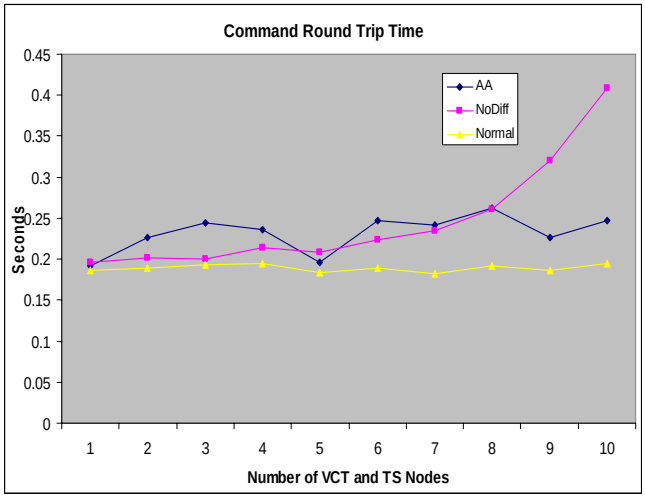

Figure 12. Command Round Trip Time

The RTT for Command data is shown in Figure 12. Similar conclusions can be drawn from this data as have been discussed previously. This data also exhibits greater variability than Normal, and better overall performance than NoDiff. The behavior of AA closely resembles that of AA in Figure 11 which reinforces the conclusion that the built-in hysteresis of the algorithm does not fully accommodate all network configurations.

Overall, the Adaptive Algorithm is able to meet its goals of being a distributed way to dynamically manage priority and still perform similarly to static prioritization, and better than no differentiation at all. The self correcting characteristic of the algorithm leads to increased variability in performance which in some cases, such as extreme network load, causes the Adaptive Algorithm to perform significantly worse than static prioritization. On the other hand, network throughput is improved when the network is under-utilized because the Adaptive Algorithm is only engaged when performance requirements are not being met. When viewed as a trade-off between simplicity of deployment, and performance benefits over non-prioritized networks, the Adaptive Algorithm can provide significant value to network architects.

9. CONCLUSIONS AND FUTURE WORK

We have presented an extension to 802.11e that dynamically adapts the contention parameters to meet performance requirements of lunar communications. We have explored the capacity of 802.11g using dynamic 802.11e to handle VoIP, command, telemetry, and video according to their respective QoS requirements. The data presented here shows that the Adaptive Algorithm can be deployed in these scenarios to effectively manage priority without a complicated centralized infrastructure. This capability will be essential for short-range lunar communications in that quick deployment will be a primary goal. The prevalence of 802.11 hardware and the possibility of the 2.4GHz band to perform at long and short ranges make it a viable option for future lunar networks. However, priority will be a necessity especially if the network carries non-critical traffic alongside mission-critical VoIP and data.

Future work on the algorithm will include improvement to the tolerance mechanism to reduce added contention from over-adjustment. It is possible that more testing with more conservative settings will result in less variable network conditions at the expense of faster convergence.

Although we feel that the algorithm presented here is important, we believe it is not the total solution. It has been shown in previous works that in general 802.11e increases contention in the network, which results in overall lower throughput. The Adaptive Algorithm does alleviate this disadvantage for some scenarios, but not all. Other works show that 802.11e under the HCCA can solve the problem of throughput at the expense of simplicity, and rapid deployment. Therefore, we expect that many 802.11e deployments will use a mix of pure EDCA and HCCA in addition to using EDCA as a backup measure for HCCA. Having an efficient and simple way to use ECDA will provide network architects with greater flexibility and will further reduce dependence on proprietary network design as we move towards expansion beyond the Earth.

10. ACKNOWLEDGEMENTS

The research described in this paper was carried out at Clemson University under the South Carolina Space Grant Consortium Research and Education Awards Program (REAP) and at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

11. REFERENCES

[1] ANSI/IEEE Std. 802.11b, 1999 Edition. IEEE Standard for Information technology – Telecommunications and Information exchange between systems – Local and metropolitan area networks – Specific requirements. Part 11. IEEE, 1999

[2] ANSI/IEEE Std. 802.11e, 1999 Edition. IEEE Standard for Information technology – Telecommunications and Information exchange between systems – Local and metropolitan area networks – Specific requirements. Part 11, Amendment 8. IEEE, 1999

[3] Wi-Fi CERTIFIED for WMM – Support for Multimedia Applications with Quality of Service in Wi-Fi Networks. Wi-Fi Alliance. September 1, 2004.

[4] Swaminathan, Arvind and James Martin. 2006. "Fairness Issues in Hybrid 802.11b/e Networks". Consumer Communications and Networking Conference. 2006 3rd IEEE Vol. 1, Issue 8-10. 50-54.

[5] Wiethölter, Sven and Hoene, Christian. 2003. Design and Verification of an IEEE 802.11e EDCA Simulation Model in ns-2.26. TKN Technical Report Series. Technical University Berlin. November, 2003.

[6] Cali, F; M. Conti; E. Gregori. 2000, "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit". IEEE/ACM Transactions On Networking, Vol. 8, No. 6, December 2000, 785-799.

[7] Cali, F; M. Conti; E. Gregori. 2000, "IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism". IEEE Journal on Selected Areas in Communications. Vol. 18. No. 9 (September).

[8] Chen, D; S. Garg; M. Kappes; K. Trivedi. 2002. "Supporting VBR VoIP Traffic in IEEE 802.11 WLAN in PCF Mode". Center for Advanced Computing and Communications, ECE Department. Duke University, Durham, NC. Avaya Labs Research, New Jersey.

[9] Siris, Vasilios and Stamatakis, George. 2006. "Optimal CWmin Selection for Achieving Proportional Fairness in Multi-Rate 802.11e WLANs: Test-bed Implementation and Evaluation". International Conference on Mobile Computing and Networking: WiNTECH 06.

[10] He, Dajiang and Charles Shen. 2003. "Simulation Study of IEEE 802.11e EDCA". Vehicular Technology Conference, Spring 2003. Vol. 1. 685-689.

[11] Xiao, Yang; L. Haizhon; S. Choi. 2004. “Protection and Guarantee for Video and Voice Traffic in IEEE 802.11e Wireless LANs”. INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communication Societies. Volume 3. Issue 7-11. 2152-2162.

[12] W. Spearman, J. Martin, "A Distributed Adaptive Algorithm for QoS in 802.11e Wireless Networks", Proceedings of the 2007 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'07), (San Diego, CA, July 2007), pp. 379-386.

[13] Lim, L.W. Malik, R. Tan, P.Y. Apichaichalermwongse, C. Ando, K. Harada, Y. Panasonic Singapore Labs. “A QoS Scheduler for IEEE 802.11e WLANs”. Consumer Communications and Networking Conference, 2004. pp. 199-204.

[14] V. Vleeschauwer, J. Janssen, G. Petit, and F. Poppe, "Quality bounds for packetized voice transport," Alcatel, Tech. Rep., 1st Quarter 2000.

[15] ITU – Series H, Audiovisual and Multimedia Systems, Infrastructure of audiovisual services – Coding of moving video, H.264. 03/2005. International Telecommunication Union.

12. BIOGRAPHY

Jay Gao joined the Jet Propulsion Laboratory in 2001 and is currently a senior research staff in the Communications Networks Group in the Telecommunication Research and Architecture section. His research is primarily focused on space-based wireless communications and networking, with emphasis on applications for the Mars Network. He is currently conducting research for developing quality-of-service (QoS)-aware protocols for the envisioned Interplanetary Network (IPN) and study optimization and protocols for deep space Ka-band communications. He also supports requirements definition and interface design activities for the Department of Defense’s Transformational Communications MilSatcom project and system engineering effort for NASA’s Exploration System and Mission Directorate (ESMD), supporting the Constellation Program for return of human to the Moon and Mars. Other research interests include optical-based sensorweb, discrete event simulation of distributed communication/sensor systems, energy efficient routing and self-organization algorithm for cooperative signal processing and sensor networks. He received his B.S., M.S., and Ph.D. degree in Electrical Engineering from UCLA in 1993, 1995, and 2000, respectively.

Will Spearman is a Master’s Candidate at Clemson University’s School of Computing. His work focuses on QoS in 802.11e and wireless networks. His background includes a B.S. in Psychology with a minor focus in Computer Science. He currently is employed at Network Appliance, Inc.

Dr. Jim Martin is an Assistant Professor in the School of Computing at Clemson University. His research interests include broadband access, autonomic computing, Internet protocols and network performance analysis. He has received funding from NASA, the Department of Justice, BMW, IBM, and Cisco. Dr Martin received his Ph.D. from North Carolina State University. Prior to joining Clemson, Dr Martin was a consultant for Gartner, and prior to that, a software engineer for IBM.