Dynamic Traffic Prioritization in 802.11e Networks

William Spearman, James Martin, James Westall

Abstract— **The IEEE 802.11 family of standards defines a collection of widely used local wireless network technologies. Early versions of the standard included no mechanism for providing differentiated services to high priority traffic such as voice and data that are sensitive to jitter, delay, and loss. The IEEE 802.11e standard provides enhancements designed to 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 and consistent control of performance levels for all priorities or simple deployment. In this work, a framework for dynamically optimizing 802.11e contention parameters is presented. Our Adaptive Algorithm is a fully distributed technique that extends 802.11e's Enhanced Distributed Channel Access.**

Index Terms—**802.11e, differentiated services, wireless networking,**

I. Introduction

In 1997 the Local Area Network/Metropolitan Area Network Committee of the IEEE released the 802.11 standard which defined a wireless standard for data networks in the 2.4 GHz radio frequency range. The original specification defined communications at a maximum data rate of 2 Mbps, but it was soon superseded by two amendments called 802.11a and 802.11b that defined maximum data rates of 54 Mbps and 11 Mbps, respectively. Although 802.11a had higher throughput than 802.11b, 802.11b was the first to market and gained the greatest market share. The two standards were not compatible due to differences in their physical layer implementations, and hardware that supported both was more expensive than simply supporting 802.11b. As a result 802.11a lost traction and 802.11b and its enhanced version 802.11g, have become the most widely used wireless communication technologies in the world.

Applications, such as streaming video, audio, or VoIP, that have specific bandwidth, latency, or jitter requirements do 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 [1].

The IEEE 802.11e standard provides enhancements that allow traffic with specific quality of service (QoS) needs to be differentiated from best-effort traffic. While these enhancements have been shown to improve throughput, latency, and jitter for high priority traffic, they do not offer precise control of performance levels. For example, strict prioritization might meet the service requirements of high priority traffic at the expense of starvation of low priority traffic.

Our research objective is to develop a class of distributed algorithms for dynamic priority management in 802.11e. Such algorithms operate by continually monitoring some performance metric and periodically adjusting the contention parameters in a way designed to drive the observed performance toward the performance objective. In this paper we present the results of our initial attempt at this undertaking. Although it is clearly only one instance of a large class of possible algorithms, the algorithm reported upon here is called the "Adaptive Algorithm." Our results to date have been mixed, and we can certainly not declare the problem solved, but we believe the work provides a useful foundation upon which improved techniques can be constructed.

The remainder of the paper is organized as follows. In Section II we present essential elements of the 802.11 MAC layer. Section III contains a description of related work. The simulation environment is described in section IV, and our Adaptive Algorithm for contention parameter adjustment presented in section V. Results of our initial testing following

Manuscript received April 1, 2008.

William Spearman was with the Clemson University. He is now with Network Appliance, Inc.

James Martin is with the School of Computing, Clemson University, Clemson SC 29634 USA. (e-mail: jim.martin@clemson.edu).

James Westall is with the School of Computing, Clemson University, Clemson SC 29634 USA. (e-mail: westall@clemson.edu).).

in Section VI and we conclude in Section VII.

II. The 802.11 MAC PROTOCOLS

A. Operational modes

The 802.11 MAC layer supports two modes of operation: the Distributed Ccoordination Function (DCF); and the Point Coordination Function (PCF). In DCF mode access to the medium is controlled by a distributed contention-based algorithm known as carrier sense multiple access with collision avoidance (CSMA-CA). In contrast, PCF is a contention-free mode in which access to the medium is granted by polls from a central access point node (AP). Both modes have characteristic advantages and disadvantages, but for a variety of reasons PCF is not widely used and will not be considered further in this paper.

B. CSMA-CA

The CSMA-CA protocol is a slotted listen-before-transmit protocol whose complexities preclude inclusion of a detailed description. The essential elements of its operation are summarized in the following. After a successful transmission, there is a mandatory interframe delay called the Short Interframe Spacing (SIFS) that provides time for the sender and receiver to switch their radios between transmit mode and receive mode. Following the SIFS exactly one station is eligible to transmit, and so no collision can occur. The receiver of a request to send (RTS) can send a clear to send (CTS); the receiver of CTS can send data, and the receiver of data may send an acknowledgement (ACK). If a second time window called the PCF interframe spacing (PIFS) passes, and no station has initiated a transmission, then the AP in a PCF network may transmit a poll. If a third time window called the DCF interframe spacing (DIFS) passes, and still no transmission has been initiated, then a slotted sequence of contention based transmission opportunities commences.

Each station has a *back off counter* that controls contention slot selection. As contention slots pass without a transmission being initiated, the back-off counter of each contending station is decremented. When a station's counter reaches 0, the station (depending on configuration) may transmit a RTS or a data packet. The back off procedure is governed by three other state variables in each station: the minimum, current and maximum sizes of the contention window (CWmin, CW, and CWmax). When new a packet arrives for transmission the CW is set to CWmin, and the back off counter is randomly chosen from [0, CW]. Collisions are inferred from failure to receive CTS or ACK. After each collision, the CW is doubled and a new back off counter is chosen. When the CW reaches CWmax, it is latched to CWMax

If the medium becomes busy while a station is conducting the count down, the station must halt its count down and defer transmission. When contention slots resume following the next DIFS, the count down resumes with the back off counter having the value it held when the count down was suspended.

It is clear that by assigning different values of CWmin and CWmax to different stations or traffic classes that some form of differentiated services may be provided. Nevertheless, the original 802.11 standards did not address this possibility.

C. Differentiated services in 802.11e

The 802.11e standard of 2005 introduced enhancements, referred to as the Hybrid Coordination Function (HCF), to both DCF and PCF. 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.

In EDCA, new parameters extend DCF to provide differentiated service to each traffic class. The new parameters are the Arbitration Inter-Frame Space (AIFS), the persistence factor (PF), and the Transmission Opportunity (TxOp). The CWmin and CWmax parameters continue to operate as previously described. The AIFS parameter is a *minimum* value for the back off counter (the back off counter is now randomly selected from [AIFS, CW]). The persistence factor is the multiplier applied to the CW when a collision occurs. The TxOp limits the amount of data that one traffic class may send while others in the same station are waiting. All parameters are given default values at each station for each traffic class, but they can be overridden by an access point.

The MAC layer queue structure in EDCA is composed of four DCF queues. Each of the four defined traffic classes, (TC0, TC1, TC2, and TC3 with TC0 having highest priority), is assigned to a unique DCF queue. Each queue has also its own set of contention parameters (AIFS, Persistence factor, TxOp, CWmin, and CWmax). The packet at the head of each queue is called the *transmission packet.* Each transmission packet has a back off counter that is managed according to the parameters of its associated queue. When the back off counters of more than one queue reach zero simultaneously, transmission packet from the highest priority traffic class is selected for transmission and the other queues experience a virtual collision. This virtual collision is treated as if a real collision has occurred. The contention window for each

colliding queue is multiplied by the persistence factor and a new back off counter is picked.

D.Providing QoS guarantees with 802.11e / EDCA

The new parameters make it easy to establish absolute priority of one traffic class over another. Nevertheless, it is a daunting (if not impossible) task to derive, a priori, contention parameter sets that can provide targeted thoughput, latency, or jitter guarantees to different traffic classes. This limitation stands in contrast to 802.16 (WiMAX) networks where such service objectives can be prescribed in a straightforward way.

As noted in the introduction, this fact motivates our research. Its thrust is to develop mechanisms that can dynamically adjust contention parameters based upon the difference in observed performance from performance objectives in such a way that target performance objectives are realized.

III.Related Work

Previous work in the 802.11 protocols has shown network parameters can be adapted to maximize overall throughput based on current network conditions [2][3]. It has also been shown that parameter tuning within existing 802.11 designs can better provide services such as VoIP some level of QoS guarantees [4]. The fact that these approaches do not accommodate multiple priority levels or standardized parameters led to the development of 802.11e. Work has been done to dynamically adjust 802.11e parameters to provide better performance and QoS. These developments, and their relevance as the basis of an Adaptive Algorithm, are discussed in the remainder of this section.

In [5] the effects of various contention window sizes are explored. The authors show that the default value of CWmin in 802.11 leads to under-utilization of the network. They showed that in networks with small numbers of stations small CWmin values greatly increased the collision probability, reduced overall throughput, and increased delays. However, as the number of stations increased and the network became fully utilized the penalty was diminished.

In [6] 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. They also show that 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. These optimizations are evaluation in a test-bed under realistic conditions.

In [2][3] the authors evaluate a mechanism that allows dynamic tuning of the timing used in the back-off algorithm in 802.11. They show 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. Their findings show that static network parameters lead to under-utilization of the medium and show the importance of a dynamic algorithm.

In [7], it is shown that the extra intra-station contention layer in 802.11e increases the possibility of collisions and increases delay. The results presented show that throughput in 802.11e is decreased and latency is increased when compared to 802.11a networks due to this extra contention. It is proposed that dynamic and aggressive tuning of the network parameters is required to achieve benefit from prioritization.

In [8] a method for dynamically tuning 802.11 using admission control and service differentiation is described. The admission control portion of the method is intended to keep the network stable when a new source is introduced. To achieve service differentiation, the authors modify the Access Point to operate on an intelligent round robin approach to TxOp allocation. The optimizations show reduced loss rates, and increased throughput.

In [9], 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. Combinations of all four contention parameters (CWmin, CWmax, persistence factor, and AIFS) are tested. Not surprisingly, it is shown that the AIFS parameter is the most effective parameter for protecting high priority traffic from background traffic. However, the authors show that using the persistence factor 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 [10] is most similar to our research. The authors use a two-level approach to providing 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 differs from our Adaptive Algorithm in that it adjusts the contention parameters on all successes and all failures while we used averaged measures of network performance..

Studies to improve specific types of traffic using tuned 802.11e parameters have been explored in [11] and [12]. In [11] fairness and throughput are improved by prioritizing based on traffic type and traffic direction. They observe that when many stations transmit, but all receive from a single access point, the access point becomes a point of congestion since it will get an unfair portion of transmission opportunities. In [12], the contention parameters are used to prioritize TCP ACK packets. Since lost ACK packets result in retransmissions, and therefore more packets on the network, ACKs are given higher priority than data packets. Not only does this approach improve network utilization, it is possible to differentiate based on this method.

IV. The Simulation environment

The *ns2* simulator version 2.28, with an EDCA add-on [12], was used to simulate an 802.11e network with the topology shown in Figure 1. Each wireless node is connected to a single base station which is in turn connected to a wired router (W0). W0 is then connected to W1, the destination router that hosts the common destination for all sources. For simplicity, the diagram shows a single source at each station, but a station can have multiple traffic sources. The wireless stations use a data rate of 54 Mbps based on the data rate and physical characteristics of homogenous 802.11g, while the wired stations use 802.3 at 100 Mbps.

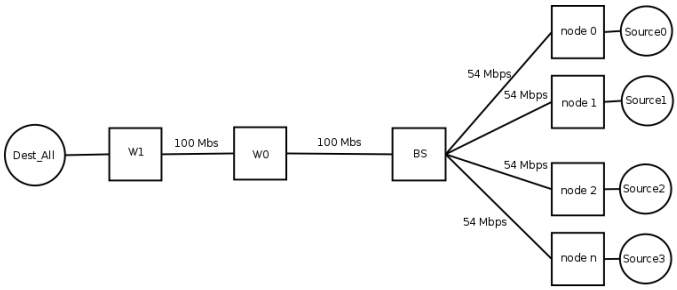


Figure 1: Simulated Network

Table I shows the traffic class contention parameter values used in the simulations. These are taken from the 802.11e specification and are the defaults for the EDCA add-on for *ns-2*. TC0 represents the highest priority, while TC3 represents the lowest priority. TC2 values are considered the default values for traffic that is unclassified, therefore for simulations where the Adaptive Algorithm is used, TC2 values are the initial parameter values.

TABLE I
Traffic Class Properties

TC / Param	TC0	TC1	TC2	TC3
AIFS	2	2	3	7
CW_{min}	7	15	31	31
CW_{max}	15	31	1023	1023

Table II shows the traffic characteristics of the data flows used in the simulations. The data flows were chosen to provide a mix of traffic types and are based on reasonable voice, video, and data profiles. 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. High priority data is also modeled as TCP/Reno CBR traffic with Delayed ACKs transmitting at a data rate of 13 Kbps, while low priority data has a data rate of 248 Kbps using UDP CBR traffic.

Standard Definition and High Definition 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 quality, but 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. Both SD and HD video are VBR traffic sources based on a Pareto distribution. During the on period having a mean value of 5 seconds, the model transmits a burst of data at a configured maximum burst rate, and ceases to transmit during the off period with mean 1 second. The Pareto model's shape parameter was set to 1.4.

The ns2 EDCA code was extended to allow data collection at the MAC layer. The MAC access delay is the performance metric used by the Adaptive Algorithm to adjust contention parameters. This is the time from a packet's entrance into the MAC queue until the time it is successfully sent.

Other performance measures are also captured in order to evaluate the operation of the network and the effectiveness of the Adaptive Algorithm. These measures include end-to-end throughput, transport level delay, jitter, and round trip time (RTT). The metrics used for each traffic type are shown in Table II. The abbreviations, TJD and RTT, represent throughput, jitter, delay (one-way latency) and round trip time.

For the results reported here each station only includes one data stream for each traffic class, and therefore a maximum of four data streams per station. While this may not be completely realistic, the data streams modeled in this research attempt to encompass a wide variety of data profiles that could be found at a single station.

TABLE II
SIMULATED WORKLOAD CHARACTERISTICS

Type / Attribute	Voice	High Priority Data	Low Priority Data	SD Video	HD Video
Protocol	UDP	TCP/ Reno	UDP	UDP	UDP
Profile	CBR	CBR	CBR	VBR	VBR
Rate (kbps)	8	13	248	4000	12000
Packet (bytes)	20	1000	1000	2048	2048
DF_TARGET	1.5	2.5	5.0	20.0	20.0
Normal Priority	TC0	TC1	TC2	TC3	TC3
Metrics	TJD	RTT	TJD	TJD	TJD

V. THE ADAPTIVE ALGORITHM

The Adaptive Algorithm tunes the AIFS, CWmin, and CWmax parameters based on locally captured performance metrics. For the results reported here, we use an indirect measure of network performance. Performance is characterized by a measure of the ratio of the recent average access delay (MAC layer queuing time) to an estimate of the minimum or *optimal* access delay. This measure is called the *delay factor*.

During the simulation, the delay factor is periodically calculated for each traffic class TC over a pre-defined time interval. Each traffic class has a fixed target for delay factor called the DF_TARGET (shown in Table II.) The objective of the Adaptive Algorithm is to drive the delay factor toward DF_TARGET by periodically adjusting the contention parameters (AIFS, CWmin, and CWmax).

The optimal mean access delay is measured whenever a new traffic class comes online at a given station and is periodically recomputed thereafter. During the brief optimal delay measurement phase AIFS, CWmin, and CWmax are minimized and the station/traffic class pair assume highest priority in the network to get a "best case" measurement of queuing delay. This measurement is not intended to measure any physical or link layer capabilities, and therefore does not include ACK times.

After the optimal discovery phase has completed, normal operation begins for the traffic class. For each station and traffic class normal operation consists of periodic measurement intervals followed by possible adjustments to the contention parameters.

During the measurement interval total delay and packet count are captured for each traffic class. At the end of the measurement interval the delay factor is computed by dividing the average delay by the optimal delay.

The goal of the adjustments is to keep the delay factor acceptably close to the target delay factor. Actual adjustment of the parameters is controlled by a secondary factor called the *delay ratio* which is computed by dividing the observed delay factor by the target delay factor (DF_TARGET). The mapping of delay ratio to parameter adjustment action is given in table 3. If the delay ratio is near 1.0, then no change is needed. If it is moderately less than 1.0 in the range (0.5, 0.8) then traffic class is performing better than its target and the contention parameters should be adjusted so as to make the behavior of the class moderately less aggressive. If the delay ratio is less than 0.5, then the the traffic class should be made significantly less aggressive. Analogous adjustments are made with the delay ratio is too high.

To significantly decrease aggressiveness, an attempt is made to numerically *increase* the contention parameters in the order (AIFS, CWmin, CWmax). If AIFS is not already latched to its maximum value of seven, it will be incremented

by one. Otherwise if AIFS is already seven, but CWmin has not yet reached 31, then CWmin will be incremented by one. If both AIFS and CWmin are latched to their maximum values then an attempt will be made to increment CWMax by 64. Moderate decreases in aggressiveness are effected by numerically increasing the contention parameters in the order (CWMax, CWmin, AIFS). To decrease the aggressiveness of a flow the contention parameters are decremented in an analogous way.

TABLE III
CONTENTION PARAMETER ADJUSTMENT

Delaly Ratio	< 0.5	0.5 – 0.8	0.8 – 1.2	1.2 – 2.0	> 2.0
Change Performed	>>	>	==	<	<<

To provide additional stability and resistance to short term transient behavior, a window based tolerance mechanism was added. The tolerance window defines the number of observations that must be inside or outside the acceptable range to trigger a parameter change. As long as the delay factor remains in the acceptable range, the tolerance window gradually increases. When the delay ratio has been in the acceptable range for a period of time, meaning the tolerance is high, the delay ratio must be out of the acceptable range multiple times to trigger a change. Conversely, when the delay ratio has been out of the acceptable range for a period of time, meaning the tolerance is low, the delay ratio must be in the acceptable range multiple times in order to guarantee that an adjustment will not be made the next time an unacceptable delay ratio is computed.

VI. RESULTS AND DISCUSSION

A. Contention parameter impact

The objective of this study was to evaluate the impact of using only one of the three contention parameters at a time to provide differentiated service. In this study, three simulations were performed using six stations, each with a single CBR UDP source with a data rate of 4Mbps. In each simulation, one station's source used the high priority TC0 settings shown in Table II for settings of contention parameters while the other five stations used the low priority TC3 settings. For example, in the AIFS simulation the TC0 senders used AIFS = 2 while the TC3 senders used AIFS = 7, but all senders used CWmin = 31 and CWmax = 1023. The Adaptive Algorithm was *not used* in this study .

Table V contains the parameter settings and results from these simulations. Throughput results are reported in Kbps, while time results are in milliseconds. Columns 2 and 3 contain the results obtained when the AIFS, CWmin, and CWmax parameters were used to differentiate traffic classes are shown in columns (2,3), (4,5), and (6,7) respectively.

TABLE IV
CONTENTION PARAMETER EFFECTS

	AIFS		Cw_{min}		Cw_{max}	
	High	Low	High	Low	High	Low
AIFS	*2*	*7*	3	3	3	3
Cw_{min}	31	31	*7*	*31*	7	7
Cw_{max}	1023	1023	1023	1023	*31*	*1023*
Thrpt	3820	3174	3940	3321	3629	2994
TptTot	19692		20546		18600	
TptDif	645		618		635	
Jitter	1.2	3.8	1.1	3.5	3.3	5.8
Delay	6	450	6	412	30	389

In [9] contained it was found that AIFS was the most effective parameter for prioritization from the perspective of the high priority data; CWmax was the best parameter from the perspective of the low priority nodes; and CWmin was the best compromise between the two. In contrast, we found that, use of CWmin gave better overall network utilization as is evidenced by the total throughput and a smaller difference between the two classes. Additionally, CWmin provided lower jitter and delay than AIFS for both low priority and high priority data. It was found that CWmax was indeed better for low priority nodes due to less aggressive prioritization of the high priority data.

CWmax adjustment provides the least impact one performance due to the fact that is does not come into play until after AIFS and CWmin in the back-off algorithm of 802.11e. The first action a queue takes when it wishes to transmit is to wait an AIFS. Therefore, small AIFS intervals affect every transmission including successful transmissions, failed transmissions, and collisions. Small CWmax values only benefit a queue when there have been multiple collisions and the CW has increased, wherein and the CW can be potentially very large. A queue with a small CWmax setting will usually have a smaller CW since the queue will reach the maximum value more quickly. Despite giving priority to some nodes, CWmax values being small also contributes to increased contention in the network since back-off windows grow to smaller sizes which results in increased probability that two stations will have zero back-off counters simultaneously. Smaller values also mean that in a crowded

network, retry limits will be reached more quickly, increasing failures and retransmits. For these reasons, CWmax is least harsh for the low priority data, least beneficial to the high priority data, and most harmful to the network.

CWmin values control the beginning value of the CW, which is the upper bound of the random window from which the back-off counter is chosen. After a queue waits an AIFS and senses the medium to be free, it decrements its back-off counter which begins set to a random value between 0 and CWmin. A queue with a smaller CWmin value is likely to be able to transmit before one with a larger value since it is more likely to have a smaller back-off counter.

It is possible the difference between the results presented here and those presented in [9] is a result of different parameter values used in the simulations. The parameter settings used here are taken from the 802.11e specification that was not finalized at the time of publication of [9]. Despite these results, AIFS was still chosen as the primary parameter for adjustment in the Adaptive Algorithm since it is the primary parameter identified by the majority of previous studies.

B.The traffic burst simulation.

The second scenario simulated introduction of a high priority traffic burst into a network operating in a steady state and carrying low priority traffic. This simulation was designed to show the Adaptive Algorithm's ability to hold a moderately loaded network in a steady state, accept a new high priority source, provide it higher quality service, and then return to a steady state when the burst completed. A CBR version of the SD video traffic source was used for each Traffic Class. TC1, TC2, and TC3 traffic sources began at time 1 and ended at time 300, and a TC0 source began at time 100 and ended at time 200. Each traffic class generated an offered load of 7 Mbps. During the non-burst period the total requested bit rates of the data streams was 21 Mbps, and during the burst period the total of the bit rates was 28 Mbps. All other network characteristics were taken from Table II.

The results are summarized in Table V and Table VI. Table V shows performance data from the burst period between simulation time 100 and 200. Table VI shows the data from the end of the burst to the end of the simulation at time 300.

TABLE V
BURST PERFORMANCE DATA

Traffic Class	TC0	
Prioritization	AA	Static
Throughput	6984.93	6953.28
Delay	12.634	5.461
Jitter	0.583	0.583
Traffic Class	TC1	
Prioritization	AA	Static
Throughput	6999.97	6790.63
Delay	6.296	6.07
Jitter	0.799	0.88
Traffic Class	TC2	
Prioritization	AA	Static
Throughput	4143.51	5499.1
Delay	380.696	285.19
Jitter	3.28	1.735
Traffic Class	TC3	
Prioritization	AA	Static
Throughput	3984.92	2699.13
Delay	396.843	537.559
Jitter	3.53	6.776

The results from time period from 0 to 100 are essentially identical to the latter third and are not shown. For comparison, simulations were also run with static prioritization and the results reported in the column labeled Static.

. TABLE VI
POST BURST PERFORMANCE DATA

TC	TC1		TC2		TC3	
Priority	AA	Static	AA	Static	AA	Static
Thpt	6999.	6860	6986	6627.	6981.	6906.
Delay	5.17	5.16	15.5	4.97	20.47	5.60
Jitter	0.21	0.11	0.40	0.14	0.4	0.16

The data from the burst period shows that the Adaptive Algorithm performs similarly to static prioritization with some differences of interest. During the post-burst period the Adaptive Algorithm sustains an aggregate throughput of 20,967 Kbps, a modest improvement of 3% over the 20,395 Kbps obtained with static parameterization. During the burst period the aggregate throughputs are 22,113 Kbps and 21,942 Kbps for adaptive and static, respectively.

Despite higher total throughput channel utilization, the Adaptive Algorithm does not deliver as low delay or jitter as static prioritization. Since the Adaptive Algorithm's most aggressive contention parameters are the TC0 values, the Adaptive Algorithm cannot be expected provide better delay and jitter values than the Static method.

During the burst, the Adaptive Algorithm also did not differentiate TC0 traffic from TC1 and TC2 traffic from TC3 nearly as well as did static provisioning. This indicates that improved choices of target delay factors are likely needed.

It was also observed that the Adaptive Algorithm does not react quite as quickly as does static when adjusting to new network conditions. When the TC0 data stream is introduced, prioritization is delayed by the time required to measure a new optimal delay, and to correct parameters based on the new measurements. It is possible to minimize these reaction periods by shortening the interval between optimal delay sampling, but at the risk of making the network less stable.

To further explore the ramifications of the new queue structure of 802.11e, an alternative simulation was conducted in which all four data sources resided in the same station rather than being located at four separate stations. In this setting the Adaptive Algorithm was able to achieve better differentiation of the four classes.

It was also observed, although not shown, that the internal queue structure within the station affects prioritization in another important way. Referring to Figure 3, when the back-off counters of two queues reach zero simultaneously, the queue with the lower queue number (TC0 < TC1 < TC2 < TC3) is given the chance to transmit and the queue with the higher value experiences a virtual collision. Therefore in a single node, multi-priority scenario even when DF_TARGET values are set to very large values amounts so that each queue will quickly adjust its parameters to TC3 values some degree prioritization occurs because of virtual collisions.

C.The Mixed Traffic Simulation

The final simulation study involved a mix of network traffic using all the traffic classes shown in Table II. Three stations were used and each transmitted VoIP, low priority data, high priority data, and one video source. The video source of the first station was high definition video (Video_HD) while the other two transmitted low definition Video_SD. Simulations were run using the Adaptive Algorithm, static prioritization, and no prioritization (802.11b/g mode). This scenario shows the Adaptive Algorithm under a realistic workload in an unsaturated network.

A summary of the results obtained with the Adaptive Algorithm is shown in Table VII. Results obtained with static prioritization appear in Table VIII, and for baseline reference, the same simulation with no prioritization is shown in Table IX. The results showed that the Adaptive Algorithm performs similarly to Static prioritization achieving almost identical throughput, delay and jitter for the VoIP traffic but slightly lower throughput for the video traffic.

The tables show that the VoIP traffic performs nearly identically under the Adaptive Algorithm and static prioritization but suffers from greatly increased delay and jitter with no prioritization. The high bandwidth low priority video data performs best with static prioritization, although the Adaptive Algorithm clearly provides better performance than is obtained with no prioritization..

In this scenario, the network was not overloaded, and prioritization was less effective since the low priority nodes did not have trouble gaining access to the medium. It is surprising that the Adaptive Algorithm obtains less total throughput than static prioritization in this scenario. This effect is mostly likely due to the use of VBR video traffic, and its effect on the stability of the algorithm. The AA is able to adapt to different network configurations, but is not able to adapt to the heavy tailed fluctuations of the video data streams.

TABLE VII
MIXED DATA, ADAPTIVE PRIORITIZATION

Traffic	VoIP	Data_Lo	Data_Hi	Vid_SD	Vid_HD
Thpt	8	248	14	3210	10206
RTT	4.8	5.8	114.2	6.9	40.2
Jitter	0.445	1.27		1.1	0.5

TABLE VIII
MIXED DATA, STATIC PRIORITIZATION

Traffic	VoIP	Data_Lo	Data_Hi	Vid_SD	Vid_HD
Thpt	8	248	14	3480	10499
RTT	4.7	5.4	106.9	6.2	83.9
Jitter	0.426	0.814		1.08	0.598

TABLE IX
MIXED DATA, NO PRIORITIZATION

Traffic	VoIP	Data_Lo	Data_Hi	Vid_SD	Vid_HD
Thpt	8	248	13.8	3326	9028
RTT	5.424	5.422	113.08	5.794	20.088
Jitter	1.063	0.95		0.801	0.485

VII.CONCLUSION

A general technique for extending 802.11e to facilitate provisioning of differentiated services has been described and the effectiveness of one implementation of the technique analyzed. The results show that dynamic optimization does lead to better overall network utilization in some cases. It was shown that queue access delays can be effectively used as performance metrics in making intelligent optimization decisions.

We feel that the Adaptive Algorithm presented here represents only the first step on the path to the solution. We believe that improved results will be obtained by incorporating other performance metrics, tuning the parameter adjustment algorithm, and choosing better performance targets.

While it is clear that the use of adaptive techniques in 802.11e will never provide the fine grained level of provisioning possible with a centralized infrastructure such as WiMAX provides, simplicity is crucial for rapid adoption and deployment in general consumer space. For this reason 802.11e will remain an important technology and therefore simple mechanisms for improving its performance will continue to be studied and eventually included in the evolving standard.

References

[1] 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.

[2] 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..

[3] 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).

[4] 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.

[5] 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.

[6] 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.

[7] He, Dajiang and Charles Shen. 2003. “Simulation Study of IEEE 802.11e EDCA”..

[8] Angelis, F. Habib, I. Altieri, P. Giambene, G. “Differentiated QoS Provision for Multimedia Traffic in WiFi Systems”. Communications, 2005. ICC 2005. 2005 IEEE International Conference on. Volume 2, May 2005. Pages 1251-1255.

[9] 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.

[10] 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.

[11] Casetti, C. Chiasserini, C. “Improving Fairness and Throughput for Voice Traffic in 802.11e EDCA”. Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium. September, 2004. Volume 1. Pages 525-530.

[12] 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.

[13]] Leith, D. Clifford, P. “Using the 802.11e EDCF to Achieve TCP Upload Fairness over WLAN Links”. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2005. WIOPT 2005. Third International Symposium on. Volume 3. Issue 7. April 2005. Pages 109–118