A Credit-based Home Access Point (CHAP) to Improve Application Performance on IEEE 802.11 Networks

Choong-Soo Lee, Mark Claypool, Robert Kinicki
Department of Computer Science
Worcester Polytechnic Institute
100 Institute Road, Worcester, MA 01609, USA
[cle01|claypool|rek]@cs.wpi.edu

ABSTRACT
The increasing availability of high speed Internet and the decreasing cost of wireless technologies has increased the number of devices in the home that wirelessly connect to the Internet. While home users often run applications with different network requirements, the applications receive the same treatment from the wireless access point (AP). It has been shown that delay sensitive applications such as VoIP, remote login and online games suffer from increased latency in the presence of throughput intensive applications such as file sharing and downloads. Typically, there are few mechanisms available at the wireless AP to mitigate these effects other than explicitly classifying traffic based on port numbers or host IP addresses. We propose a Credit-based Home Access Point (CHAP) that features a credit-based queue management technique designed to eliminate the need for explicit configuration of per-application quality. CHAP dynamically adjusts the priority of flows from different devices to better satisfy their application requirements based on the wireless conditions. Preliminary evaluation of CHAP compared with standard DropTail and Strict Priority Queuing (SPQ) shows the merits of our approach.

1. INTRODUCTION
Wireless access points (APs) have become very popular for networking multiple devices at home and allow individuals to simultaneously share one Internet connection even though running different applications. In addition to desktop and laptop computers, home users also have many other devices with network capabilities. These devices include gaming consoles (hand-held and stationary), Voice-over-IP (VoIP) phones, video streaming servers, cell phones and PDAs. Applications run on these devices often have their own Quality-of-Service (QoS) requirements. For example, a user playing a game on a game console typically needs low latency for enjoyable gameplay, while a user downloading a high-definition movie trailer prefers consistent, high throughput. Concurrent activities done over the same Internet connection can cause congestion, typically degrading application performance by increasing loss or latency or restricting bandwidth. For example, when one user is in a VoIP session while another user downloads a large file, VoIP quality can suffer from choppy sound and increases in delay that affect interactivity.

An increase in latency for delay sensitive applications indicates that the Internet connection setup, there are three places where the channel usage time as the credit cost metric for each flow. Given the bandwidth along the link between the broadband device and the ISP gateway depends on the Internet subscription. The fastest residential Internet access offered in the US is approximately 50 Mbps downstream and 20 Mbps upstream (from Verizon). However, ISPs plan to provide faster broadband access to homes worldwide. For example, Hong Kong, Japan, and the Netherlands have deployed or are testing residential connections of 1 Gbps and Sweden tested 40 Gbps connections to select homes. With such high bandwidths available to homes, it is most likely that the wireless AP will become the source of the bottleneck link. Thus, the wireless shared medium (by all devices and neighbors) is likely to be the capacity bottleneck in the long-term.

There have been solutions proposed and implemented by wireless AP manufacturers to provide basic QoS support. Some APs prioritize a physical Ethernet port for a specific device connected to that port, or prioritize flows based on their protocol type and port number. However, the average home user has difficulty understanding and configuring even basic features of a wireless AP without adding the burden of having to configure an AP with port and flow information. Even for experienced users, these solutions have limitations. For port-based priorities, users need to know the protocol and ports for the applications they run and new, unclassified applications are unable to receive appropriate treatment.

The limitations of static QoS identification has prompted research in automatic classification of flows to provide for specific QoS needs. Classification of flows removes the need for users to configure their wireless APs to prioritize their traffic. However, in addition to the limited ability to classify new applications, changes in the wireless network connectivity can bring down the performance and quality of the home network if flows with poor connectivity are given higher priority. In such a situation, it is better not to give priority to flows with poor connectivity despite their classification.

To address the shortcomings of static or dynamic flow classification and treatment while providing varied per-application QoS, this paper presents the Credit-based Home Access Point (CHAP). CHAP uses a new queue management scheme that uses per-flow credits to choose which packets to send first. The primary goal of this work is to improve overall application quality in home networks with minimal configuration and no explicit classification. CHAP takes advantage of the relationship between delay tolerance and bandwidth usage of users’ activities. This relationship is mapped such that one flow will have more credits than another flow that uses more bandwidth. In addition, flows in bad wireless locations are automatically given relatively lower priority to gracefully mitigate degraded wireless conditions. Thus, CHAP uses wireless channel usage time as the credit cost metric for each flow. Given the same wireless connectivity, flows with high bandwidth have lower...
credits compared to flows with low bandwidth. Given the same bandwidth, flows with good connectivity amass more credits than flows with poor connectivity.

CHAP is evaluated using NS-2 [6] simulations through a preliminary set of scenarios. Our simulation results show that CHAP: 1) provides application quality that is superior to DropTail in all scenarios tested; 2) yields Strict Priority Queue (SPQ)-equivalent application quality when all applications are in good locations; and 3) achieves higher overall network performance than SPQ when multimedia applications move from good to bad locations. These application quality improvements are accomplished while providing performance similar to DropTail and SPQ with respect to traditional application throughputs and overall network performance.

The rest of this paper is organized as follows: Section 2 focuses on the derivation and details of the CHAP mechanisms; Section 3 describes the set of simulation experiments run to evaluate CHAP over a range of network conditions and includes analysis of the simulation results and detailed comparisons of the performance of CHAP with DropTail and SPQ; and Section 4 summarizes our findings and considers further evaluation, extensions and future work.

2. PROPOSED APPROACH

2.1 Bandwidth and Delay for Residential Activities

While there is a wide variety of applications that users run in their home, such applications can be broadly grouped based on like activities. Popular user activities that use network applications include Web browsing, audio and video streaming, Voice over IP (VoIP), instant messaging, online games, email, and file downloading. Furthermore, the wide range of network applications supporting these activities can be broadly characterized based on their bandwidth usage and delay constraints. Typically, applications of a similar nature share common characteristics.

![Figure 1: Characteristics of Network Applications](image)

Figure 1 depicts the bandwidth usage and delay tolerance of the various user activities. As the figure suggests, there is a direct relationship between each activity’s bandwidth usage and its delay tolerance, indicated by the tilted gray rectangle. The more bandwidth an activity uses, the more delay tolerant it is and vice versa.

2.2 CHAP Algorithm

Typical operating systems schedulers favor processes that are not CPU-bound, typically those that are interactive. Our proposed approach, Credit-based Home Access Point (CHAP), draws upon the similarity of process scheduling to packet transmissions scheduling for each flow, exploiting the relationship between delay tolerance and bandwidth usage shown in Figure 1. Bandwidth usage can be observed easily at the AP because every packet that arrives and leaves the home network goes through the AP. Since the AP is responsible for routing traffic to the correct host within the home network, the typical AP keeps a Network Address Translation (NAT) table with a 5 tuple per entry: source address, source port, destination address, destination port, and protocol. This same information set can be used to identify flows. Additionally, CHAP keeps per-flow credits to aid in scheduling with the goal of improving overall application quality.

![Figure 2: Block Diagram of CHAP](image)

Figure 2 depicts the components of CHAP inside an AP. The solid arrows show the downstream and upstream network traffic in and out of the home network. The major difference between CHAP and traditional APs is that in CHAP the downstream and upstream queues get control information from the enhanced NAT component with credits to prioritize the packets in the queue. In addition, the NAT component requires control information from the 802.11 data link layer to update the credits.

To adjust credits, CHAP uses the channel usage time at the wireless data link layer which reflects the combination of its bandwidth usage and wireless connectivity. The more frames a flow transmits, the fewer credits it has. The more time the frames take to transmit, the fewer credits the flow has. Therefore, in general, flows that drain credits slower have higher priority. This mechanism ensures nodes with poor connectivity, even if they have low application-level bandwidth (i.e., delay sensitive), do not degrade performance for the entire network.

In addition to decrementing credits during transmission, CHAP boosts credits when all flows run out. The basic form of credit boost is based on an earlier incarnation of the Linux process scheduler: 

$$\alpha_i' = \alpha_i + I$$

where $i$ is the flow index, $\alpha_i$ is the credit of flow $i$, and $I$ is the increment. This mechanism is used when every flow with data backlogged (in queue) has 0 (or fewer) credits and is applied to every active flow. The parameter $I$ is in units of time, consistent with the unit of credits.

Algorithms 1 and 2 summarize the dequeue and enqueue functions for downstream traffic in a wordy, pseudo code format. dequeue() returns the first packet that belongs to the flow with the most credits in the queue. Step 1 finds the flow with the most credits. If a credit boost is necessary in steps 2-4, CHAP boosts credits for all active flows. Step 5-6 dequeue the packet from the flow picked in Step 1. Step 7 decreases the credits according to the
Algorithm 1 CHAP::enqueue(p)

1: find \( k \) such that \( \alpha_k = \max[\alpha_1, ..., \alpha_n] \) and \( k \in \) set of backlogged flows
2: if \( \alpha_k \leq 0 \) then
3: \( \alpha_i = \frac{\alpha_i}{2} + I \) for all \( i \in \) active flows
4: end if
5: \( p = \) the first packet from flow \( k \) in the queue
6: return \( p \)
7: \( \alpha_k = \alpha_k - \text{cost} \)

Algorithm 2 CHAP::dequeue()

1: \( j = \) flow ID of \( p \)
2: \( p \) to the end of the queue
3: if the queue is full then
4: find \( k \) such that \( \alpha_k = \min[\alpha_1, ..., \alpha_n] \) and \( k \in \) set of backlogged flows
5: \( p_{tmp} = \) the first packet from flow \( k \) in the queue
6: drop(\( p_{tmp} \))
7: end if

transmission time spent at the wireless data link layer.

enqueue(p) adds the incoming packet \( p \) to the queue. If the queue is full with the incoming packet, CHAP drops the first packet that belongs to the flow with the least credits in the queue. Step 1 retrieves the flow ID of the incoming packet \( p \). Step 2 adds the incoming packet to the queue. Step 3 checks if the queue is full. If the queue is full, CHAP proceeds to find a packet to drop from the backlogged flow with the fewest credits following Steps 4-6. Steps 3-6 ensure that there is always room for an incoming packet by limiting the maximum number of packets in the queue to \( q_{size} - 1 \).

In IEEE 802.11, the wireless channel is shared by both downstream and upstream traffic for the same flow. Therefore, the credits need to reflect how much wireless resources a flow uses in both up and downstream directions. At the receiver (the AP), it is impossible to measure exactly how long an upstream transmission took without modifying the wireless protocol. Instead, CHAP keeps a exponentially weighted average of the cost of transmitting each packet along with the credit for each flow. When receiving an upstream packet, if there is an average cost for that flow, CHAP decrements the flow’s credits by the average cost. If there is no average cost, CHAP decrements the flow’s credits by an estimated cost based on the largest packet size and good wireless connectivity.

Figure 3 illustrates the simulation topology used to model a typical home network. Connections between application servers and the gateway and between the gateway and the AP are 100 Mbps Ethernet links with 1 ms latencies. Each client communicates with the infrastructure-mode AP via IEEE 802.11g. The AP sends beacon frames every 100 ms and the scanning mode is set to passive. The AP queue size is 35 packets, a typical queue length for residential wireless access points [5]. All nodes utilize DropTail queuing, except for the AP on the wireless interface. Unless otherwise noted, all wireless nodes are 1 meter from the AP. CHAP is initialized such that \( I \) and the default credits are 100 ms per flow.

Figure 5 gives the flow arrival and departure schedules. Four distinct applications are tested separately in different scenarios. The application of interest starts at 30 seconds and finishes at 330 seconds.

3.1 Simulated Applications

NS-2 includes FTP, Constant Bit Rate (CBR) and FullTCP applications. FTP applications send data over TCP connections. A CBR
The VoIP application uses the E-Model [2] which provides a MOS mapped from R-factor, a measure of audio impairments from delay and the codec. Delay degrades VoIP quality when one-way delay exceeds 177 ms. In the home scenarios simulated, delay is less than tens of milliseconds and the E-Model is unaffected. However, wireless losses decrease VoIP quality controlled by the codec. Jitter does not factor into VoIP quality as the VoIP player technology mitigates jitter effects. The E-Model MOS ranges from 1.0 (worst) to 4.5 (best).

Video application performance is gauged by the playback frame rate which has a range of 0 to the maximum encoded frame rate. In the simulations, it is assumed the receiver plays all frames when they arrive and it never discards late frames.

Web application performance is measured by response time calculated from the time the client sends out the request until the last object is completely received. Response time varies greatly due to variability in object sizes and distances to the server.

FTP application performance uses throughput. Higher throughput results in shorter download time (better performance) while lower throughput results in longer download time.

Table 1 summarizes the performance from the simulations of the four distinct applications of interest. Game, VoIP and video application performance metrics and FTP flow throughputs represent averages over the time interval between 150 and 210 seconds. Web response times are averaged over the entire duration of the simulation. Due to space limitations, only the game simulation results are analyzed in detail. Note that regardless of the application of interest, in Table 1 FTP throughputs remain essentially constant across the three choices for AP queuing implementations.

The VoIP application achieves a high MOS value of 4.37 with the DropTail queue because VoIP MOS is only degraded by packet loss. CHAP and SPQ both improve VoIP MOS by 1% to 4.42. While the video application plays at only 10 frames per second (fps) under DropTail, CHAP and SPQ both run at 30 fps, the maximum playable frame rate. This is a 201% improvement over DropTail. In the Web traffic simulation, average Web response time for DropTail queue is 109 ms. SPQ lowers Web response time by 67% down to 34 ms and CHAP provides 39 ms Web response time, a 64% improvement.

While the game application with DropTail achieves a 3.67 MOS, both SPQ and Chap AP queuing improve the MOS by 16% to 4.27. Figure 6 provides game performance results over the duration of the three simulations with the first row of graphs comparing game, FTP and total throughput for the three AP queues. Total throughputs in all three cases are about the same. CHAP provides the same bandwidth for the FTP applications but with reduced variance due to the round-robin nature of the credit scheme for flows with equal effective bandwidths. Figures 6d, 6e and 6f demonstrate that the jitter in packet arrivals for the game application has less variabili-
ity when either CHAP or SPQ are used by the AP compared with DropTail. Figures 6g, 6h and 6i show the G-Model MOS and its factors. In Figure 6g, the MOS decreases once the first FTP application starts at 90 seconds. The delay is increased due to a nearly full queue and the jitter becomes pronounced. Figures 6h and 6i show that both CHAP and SPQ provide nearly constant delay and jitter for the game application, resulting in a high MOS value.

As mentioned in Section 1, by providing pre-classified applications with a static priority, SPQ may provide the best performance when all wireless nodes are in good range. However, whenever a device running an application with high priority moves to a bad location this yields an increased number of retransmissions and loss events. Overall performance is improved by lowering the priority of the application in the bad location to provide more channel capacity to flows running from better wireless locations. Figure 7 depicts such behavior by moving the wireless node running the video application farther away from the AP in subsequent simulation runs. Namely, simulations were run with the same setup and settings except the video client was placed at 1, 5, 10, 15, 20, 25 and 30 meters from the AP. Video and FTP application throughputs are about the same for DropTail and SPQ with FTP throughputs dropping close to zero at about 15 m. CHAP starts punishing the video application as early as 10 m, but it keeps FTP throughputs higher as performance degrades gracefully all the way to 30 m. However, it is important to realize that video throughput is not the only measure of quality for video applications. Figure 8 graphs video playable frame rate across different distances for DropTail, CHAP and SPQ. Although DropTail maintains about the same throughput as SPQ, the DropTail video frame rate is far below SPQ video frame rate. The burstiness of packet drops from DropTail causes a large degradation in video frame rate. The video frame rate with CHAP starts to degrade at 10 m. However, all three video frame rates are near zero at 15 m. Therefore, assisting the video traffic beyond 15 m does not help the video quality for users. Thus, SPQ at 15 m provides no real benefit to the video while FTP gets near zero throughput. Although CHAP starts to degrade the video quality earlier, it ends up helping the FTP applications in situations where the video is unwatchable.

4. CONCLUSIONS

This paper introduces CHAP, a buffer queue management scheme designed specifically to improve application QoS over home wireless networks. CHAP retains and manages credits for all flows passing through a wireless AP. CHAP boosts credits when all ac-
pre-configured priorities for known applications. In the presence of clients in good locations, it requires
clients in bad locations, CHAP performs better than SPQ with re-
performs nearly as well as SPQ. Although SPQ provides the bes
t quality improvement over DropTail for some applica-
tions and
algorithms 1 and 2 including analysis of the setting for the incre-
ment (P2P) applications. Moreover, formal analysis of CHAP in Algo-
loops of conditions and circumstances that include a wider range
of clients in bad locations, CHAP performs better than SPQ with re-
spect to FTP throughputs while still assisting the video applica-
tion as much as is prudent.

Our ongoing work involves evaluating CHAP under a broader
range of conditions and circumstances that include a wider range
of the number of wireless clients, more variability in wireless client
locations and AP queue size, and the presence of upstream traffic
and consideration of other background traffic such as Peer-to-Peer
(P2P) applications. Moreover, formal analysis of CHAP in Algo-
rithms 1 and 2 including analysis of the setting for the increment I,
is underway. A prototype implementation using a Linux-based ac-
cess point is planned to validate the simulations and to under-
stand implementation nuances.

5. REFERENCES
[1] T. Beigbeder, R. Coughlan, C. Lusher, J. Plunkett, E. Agu,
and M. Claypool. The Effects of Loss and Latency on User
Performance in Unreal Tournament 2003. In Proceedings of
and A. Barreto. An E-Model Implementation for Speech
10th IEEE Symposium on Computers and Communications,
Quality Characterization of Single-Layer Video Streams
Encoded with H.264/MPEG-4 Advanced Video Coding
Standard and Scalable Video Coding Extension. In IEEE
Transactions on Broadcasting, volume 54, pages 698–718,
September 2008.
http://tinyurl.com/lrwe3u.
Measuring Queue Capacities of IEEE 802.11 Wireless
International Conference on Broadband Communications,
Networks and Systems (BROADNETS), Raleigh, NC, USA,
Sept. 2007.
Online at http://www.isi.edu/nsnam/ns/.
Performance Evaluation with Frame Size and Quality Traces
Ahmed. Predicting the perceived quality of a first person
shooter: the Quake IV G-model. In Proceedings of 5th ACM
SIGCOMM workshop on Network and system support for
Selecting Practical MPEG Group of Pictures. In Proceedings
of IASTED International Conference on Internet and
Multimedia Systems and Applications (EuroIMSA),
Innsbruck, Austria, February 2006.
game halo 2. In Proceedings of the International Workshop
on Network and Operating Systems Support for Digital
Audio and Video, 2005.