Uplink Traffic Control in Home 802.11 Wireless Networks

Yanlin Li
Carnegie Mellon University
yanlli@cmu.edu

Dina Papagiannaki
Telefonica Research
dina@tid.es

Anmol Sheth
Technicolor Research
anmol.sheth@technicolor.com

ABSTRACT

IEEE 802.11 wireless networks become increasing more complex and interesting inside homes. A number of home automation, home security, and entertainment products rely on wireless technologies for easy deployment without the need for wiring. Moreover, a number of such applications are fundamentally changing the traffic mix of a home wireless network, resulting in uplink traffic that is not only triggered by the users but that could potentially be nearly continuous in nature, such as wireless home security products, where each individual camera is likely to stream large amounts of data in high traffic areas.

Given the diversity of traffic sources and their importance to the user, wireless home APs today can ship with Wireless Multimedia (WMM) support that prioritizes VoIP and video traffic for better user experience. In this paper, however, we note that the type and importance of applications to a home user may be much more diverse than 4 traffic classes could accommodate. In response, we survey the landscape of possible solution in particular when it comes to pacing traffic sources inside the network. We discuss the tradeoffs that such a design space exposes and test the performance of several solutions using ns3 simulations. Finally, we note that instead of a strict prioritization of traffic streams, a simple mechanism by which the user can pace traffic to provision more resource to the traffic of importance may be sufficient.

Categories and Subject Descriptors

C.2.2 [**Computer Systems Organization**]: Computer Communication Networks—*Network Protocols*

General Terms

Algorithms, Design and Measurement

Keywords

IEEE 802.11, Uplink Traffic Control, WMM

HomeNets'11, August 15, 2011, Toronto, Ontario, Canada.

1. INTRODUCTION

Wireless Access Points (APs) have not only revolutionized the way we access computing resources and the Internet in the enterprise space, but also inside our own homes. In fact, one could argue that the advent of wireless networking in residences has been pivotal, allowing home users to use the Internet without the need for wires (given that aesthetics are far more important inside homes than enterprises).

Such wireless access, originally, was constrained to the use of WiFi to access the Internet using a laptop or a desktop. Almost ten years later, wireless technologies in the home span, WiFi, UWB, Zigbee, Z-wave, offering services like home automation, home security, entertainment and even energy management. Traffic in a home wireless network is no longer caused by a user action but also by a number of sensors, like cameras or energy meters, and may no longer be just downlink. Security cameras and energy sensors will cause uplink traffic that may range from a few Kbps up to a few Mbps, if the cameras stream in color and high resolution.

An immediate consequence of all the above is that the AP is no longer the dominant source of traffic on the network. Second, the user no longer has a clear idea of what the network resource utilization is across time. It's not unusual for automatic processes, like say a computer backup, to interfere with a user watching streamed content on his WiFi-enabled TV. The user will have no idea why the quality of the movie degraded, let alone a way to let the network know that at this point in time the computer backup should have the least priority consuming the network's resources.

Quality of service and the treatment received by different traffic sources in a WiFi network is the topic of the IEEE 802.11e working group. Wireless MultiMedia (WMM) extensions are a mechanism for VoIP and video traffic to be preferentially treated in a 802.11 network. While WMM-capable APs have been shipping for a while now, (i) preferential treatment of traffic relies on the traffic being tagged as video or voice, and is not commonly followed by application developers, (ii) the mix of home applications no longer fall in 4 categories, (iii) the importance of an application at any point in time should ultimately be defined by the home user, instead of relying on a fixed categorization, and (iv) there is significant heterogeneity in the implementation of WMM across drivers and device manufacturers. This limits the extent to which WMM can address problems in the home as we move towards more service-rich wireless home networks.

The primary question we address in this paper is, as wireless traffic in homes continue to become increasingly rich and generate high volume of uplink traffic, what are the primary

link level mechanisms at our disposal to pace wireless sources inside a home, and what are their tradeoffs. Specifically, the desired mechanism should provide fine-grained control of uplink flows without impacting performance of the network adversely and require minimal client-side support to simplify deployment.

In the rest of the paper, we provide a taxonomy of four mechanisms that could be used for such a purpose and that could be easily triggered by a home user. Our approach does not rely on the user explicitly providing a priority list of services inside the home, but on his/her decision to lower the priority of specific sources such that more resources become available to the sources that matter to him/her.

2. SOLUTION SPACE

Limiting the amount of resource consumed by a wireless source in a AP-based WiFi network can be done in a few ways. 1) At the Link Layer, the AP may decide to never send back an Acknowledgment, essentially forcing the wireless source to retransmit the packet, while increasing its back-off, 2) At the Link Layer, the AP may decide to delay sending back the Acknowledgement, hoping that the reception of the ACK packet happens before the expiration of the back-off timer, 3) The AP may use CTS frames to silence other stations, thus forcing them to limit their medium use. 4) Finally, one could envision a solution by which the AP could explicitly inform stations as to how often they should access the medium. In what follows, we look into each of these solutions in detail.

2.1 Dropping Packets

In a 802.11 wireless network, every successful packet reception by the destination is accompanied by the transmission of an Acknowledgment (ACK) packet to the original source. If the wireless source does not receive the ACK packet within a specific time period, then the source infers that the original packet transmission was unsuccessful. In response, the wireless station will double its contention window and randomly pick a number from that window to determine the number of slots it will wait before re-attempting a transmission. This process will be repeated up to 7 times, before the packet is declared lost. Table 1 shows the size of contention windows (CW).

Based on the above, it becomes evident that one could essentially pace a wireless source by artificially withholding ACKs back. That would essentially force the wireless source to back-off for longer and longer periods of time, thus injecting one packet across a much longer time interval.

Table 1: The Size of contention windows (CW) for each retransmission

Retransmission	1	2	3	4	5	6
CW (slots)	32	64	128	256	512	1024

2.2 Delaying ACKs

As described in Section 2.1, a wireless source waits for an ACK packet after each transmission. If failing to receive the expected ACK packets, the wireless source sleeps for a random number of time slots before retransmissions. If AP delays sending the ACK packet for a short period, it is likely that the wireless source can still receive the expected ACK packet before retransmissions. Therefore, AP can force a wireless source to sleep for a short period before retransmissions. Figure 1 shows the procedure of a normal transmission, and the procedure in which AP delays an ACK packet. In a normal transmission, AP sends an ACK packet back to

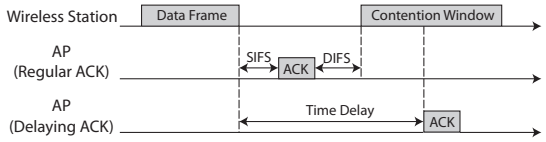

Figure 1: AP delays the ACK packet for a time period

the wireless station after receiving the data frame. In the procedure where AP delays sending the ACK packet, AP sends the ACK packet to the wireless source after a time delay, and the wireless source receives the expected ACK packet during back-off time.

Collisions may happen if AP delays sending the ACK packet in this way. One collision happens if the wireless source retransmits the packet during the time that the AP starts to send out the delayed ACK. The other collision happens if other stations send data frame to AP when the delayed ACK is sent out. If AP receives the retransmitted packet before sending out the delayed ACK packet, the time slots to delay the ACK needs to be adjusted to guarantee that there is at least a SIFS time interval between receiving the retransmitted the data frame and sending the ACK packet.

2.3 Silencing Stations by CTS frames

Request To Send (RTS) and Clear To Send (CTS) frames are used to coordinate access to the wireless channel. In detail a wireless source that has large size packets to transmit sends a RTS frame to AP to request medium. If the medium (AP) is available, AP sends a CTS frame to the wireless source to approve the request. In the CTS frame, there is a NAV time that AP allocate to the wireless station. All other wireless sources that can hear the CTS frame sleep for the NAV time after receiving the CTS frame. Figure 2 shows the protocol.

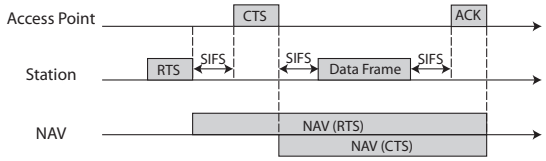

Figure 2: AP allocate time duration to a wireless station by a CTS frame

The AP can leverage the CTS frame to allocate the medium to a wireless source (the protected wireless source) by explicitly sending an unsolicited CTS frames to this source. Then, the protected source has the medium reserved until the duration field specified in the CTS frame while all other wireless sources are silenced.

Figure 3 shows the protocol in detail. In this figure, the AP sends a CTS frame to allocate a large NAV time to the protected source. During the NAV time, the protected

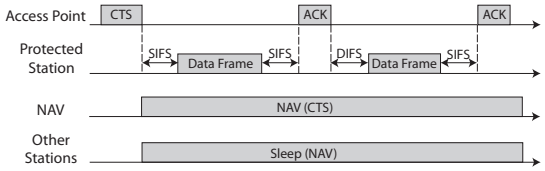

Figure 3: CTS-based protocol to allocate medium to protected stations, and silence unprotected stations

station sends data frames to the AP, while all other stations fall into sleep.

2.4 Adding Backoff Information in ACK Packets

The idea of the fourth mechanism is very simple. When AP sends an ACK packet to a wireless station, two bytes of additional backoff information (additional NAV duration) is added to the ACK packet. After receiving the ACK, the intended receiver of the ACK checks the two bytes additional NAV duration, and sleeps for the number of time slots based on the value of the additional NAV duration included in the modified ACK packets (Please note that the wireless source that the ACK packet targets checks the value of the additional NAV. All other wireless stations do not update NAV time based on the additional two bytes NAV duration). We call the modified ACK packets as NAV-ACK. Figure 4 shows the format of a regular ACK packets, and a modified ACK packet that contains 2 bytes additional NAV duration. In the modified ACK packet, the two bytes NAV Duration are added between receiver address and FCS.

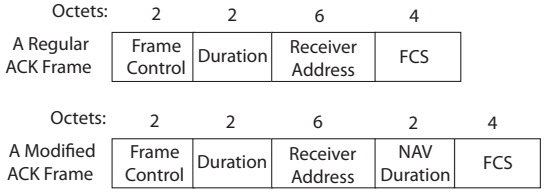

Figure 4: A regular ACK packet, and a modified ACK packet that contains additional NAV duration.

3. EVALUATION

We evaluate our mechanisms using ns3 simulator. In this section, we first describe the network topology and configuration, and then describe the evaluation results in detail.

3.1 Network Topology and Configuration

Figure 5 shows the network topology and configuration, on which we evaluate our mechanisms. As shown in this figure, $n1$ and $n2$ are wireless stations in IEEE 802.11a wireless network, while $n3$ and $n4$ are stations in a LAN network (Ethernet). An Access Point, AP, connects the wireless network with the LAN network. All the packets transmitted between the 802.11 wireless network and the LAN network must go through AP. The maximum data rate of the wireless network is 9 Mbps. $n1$ and $n2$ are static stations with constant received signal strength (RSS) 80 dB. The date rate of the LAN network is 1000 Mbps with delay 6560 nanoseconds. In our evaluation, $n1$ and $n2$ are UDP clients while $n3$

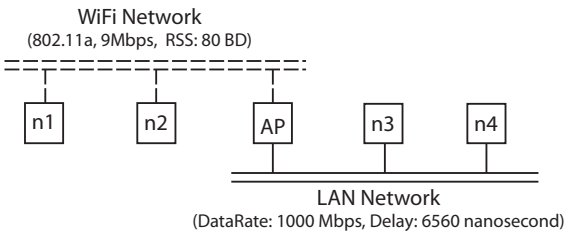

Figure 5: Network topology and configuration

and $n4$ are UDP servers. $n1$ and $n2$ continually send UDP packets (1024 bytes each packet) to UDP server $n3$ and $n4$ separately for 14 seconds. The time interval between two consecutive UDP packets sent by $n1$ or $n2$ are 100 microseconds in the application layer. Before 4 second, $n1$ and $n2$ share the medium (AP) equally. Starting from 4 second, AP begins to decrease the throughput of the uplink traffic from $n1$ using different mechanisms described in Section 2.

3.2 Evaluation Results

We implement the four mechanisms on ns3 simulator by modifying the source code of ns3 simulator, and then evaluate the four mechanisms. In this section, we describe and analyze the evaluation results in detail.

3.2.1 Dropping Packet

In the first mechanism, AP decreases the throughput of the uplink traffic from a wireless station by explicitly dropping packets from the wireless station. This increases the backoff time while other stations in the network get a higher probability to access the medium. In our evaluation, starting from 4 second AP begins to apply this mechanism to decrease the uplink traffic from $n1$. In detail, from 4 second to 6 second, AP drops the packet from $n1$ once for each link layer packet transmission. Therefore, $n1$ waits for a random backoff time that is smaller than 32 slots before retransmission from 4 second to 6 second. Every two seconds, AP increases the number of dropping packets for each link layer packet transmission by one, until reaching the retransmission threshold.

Figure 6 shows the evaluation results. The X-axis is time in seconds while Y-axis is the throughput of uplink traffic in the application layer by Mbps. The traffic flows from $n1$ is the unprotected uplink traffic, while the traffic flows from $n2$ is the protected uplink traffic. As shown in this figure, before 4 second, the throughput of $n1$ and $n2$ are the same, about 4 Mbps. Starting from 4 second, the throughput of $n1$ begins to decrease because AP starts to explicitly drop packets from $n1$, while at the same time the throughput of $n2$ increases because $n2$ gets more medium resource. Finally, by 14 second, the throughput of $n1$ is lower than 1 Mbps while the throughput of $n2$ is about 7 Mbps.

3.2.2 Delaying ACK

In this technique, the AP delays the ACK to the client such that it forces the client to increase its contention window but sends the ACK before the client can retransmit the frame. In our evaluation, starting from 4 second, AP begins to decrease the throughput of $n1$ by delaying the ACK packets that are sent to $n1$. Also AP increases the number of time slots to delay the ACK packet every two seconds. The

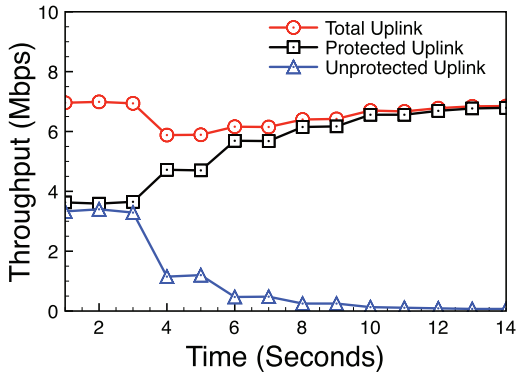

Figure 6: Throughput of uplink traffics when AP drops packets from the unprotected source.

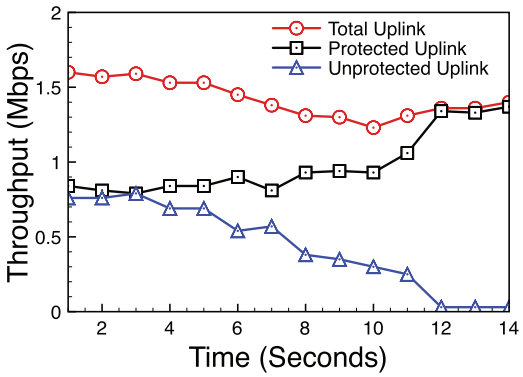

Figure 7: Throughput of uplink traffics in a 802.11b wireless network when AP delays sending ACK packet to the unprotected source.

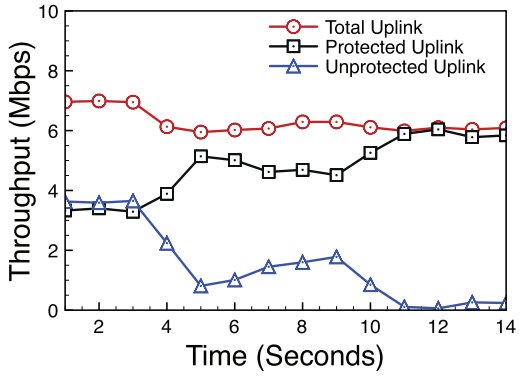

Figure 8: Throughput of uplink traffics when AP allocates medium resource by misusing of CTS frames.

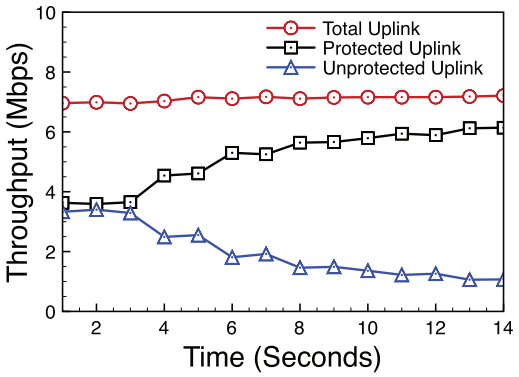

Figure 9: Throughput of uplink traffics in when AP allocate medium resource by NAV-ACK packets.

number of time slots that an ACK packet is delayed during different time periods is shown in Table 2. As discussed

Table 2: The number of time slots that an ACK packet is delayed in different time period.

Time (second)	4-6	6-8	8-10	10-12	12-14
Delay (Slots)	3	5	7	9	37

in Section 2.2, the delayed ACK may causes collision with packets sent by any wireless station in the 802.11 wireless network. To avoid collision with the retransmitted packet by the wireless station, we try to choose small time slots (i.e. 3, 5, 7, 9) or very large time slots (i.e. 37) to delay the ACK packet. However, when we evaluate this mechanism in a 802.11a wireless network, there are still many collisions. Thus, to decrease the number of collisions, we choose a "slower" network, *802.11b wireless network with maximum data rate of 2 Mbps*, to evaluate this mechanism. The evaluation results are shown in Figure 7. From 4 second to 14 second, the throughput of the unprotected uplink traffic from $n1$ decreases from 0.8 Mbps to about 0.1 Mbps, while the throughput of the protected uplink traffic from $n2$ increases from 0.8 Mbps to 1.5 Mbps.

3.2.3 *Silencing Stations by CTS Frames*

The third mechanism is silencing wireless stations by having the AP to transmit unsolicited CTS frames. In this way, the "protected" station can obtain sufficient medium resource for uplink traffic. Similar to the previous setup, before 4 seconds, $n1$ and $n2$ share the medium equally. Starting from 4 second, AP begins to allocate medium resources to $n2$ by sending CTS frames to $n2$. AP increases the NAV time duration contained in CTS frames every second. Table 3 shows that NAV duration contained in CTS frames during different time periods. The evaluation results are

Table 3: NAV time duration contained in CTS frames in different time periods

Time (second)	4-5	5-6	6-7	7-8
NAV duration (μs)	100	400	800	1200
Time (second)	8-9	9-10	10-11	11-14
NAV duration (μs)	1600	2000	2400	3000

shown in Figure 8. The X-axis is the time by seconds while the Y-axis is the throughput of the uplink traffic in the application layer by Mbps. The evaluation results show that from 4 second to 14 second, the throughput of the unprotected uplink traffic from $n1$ decreases from 2.5 Mbps to 0.1 Mbps. When the NAV duration in CTS frames is 3000 microsecond (from 11 second to 14 second), the uplink traffic from $n1$ is almost completely blocked. At the same time, the throughput of the protected uplink traffic from $n2$ increases from about 4 Mbps to about 6 Mbps. Because of the additional CTS frames, the total throughput of uplink traffics decreases from about 7 Mbps to about 6 Mbps after AP begins to send CTS frames to allocate medium resources.

3.2.4 *Adding Backoff Information in ACK Packets*

In the forth mechanism, 2 bytes additional backoff information (NAV duration) are added in ACK packets by AP to inform the wireless station to sleep for a time period after receiving the modified ACK packets. We evaluate this mechanism by adding different backoff values in ACK packets. Starting from 4 second, AP starts to send the modified ACK packets that include additional NAV duration to $n1$. Table 4 shows the value of the two bytes additional NAV duration added in ACK packets in different time periods. Figure 9 shows the evaluation results. From 4 second to 14 second, the throughput of the protected uplink traffic from $n2$ increases smoothly from about 4.5 Mbps to about 6 Mbps while the throughput of the unprotected uplink traffic from $n1$ decreases from 2.5 Mbps to 1 Mbps. As the overhead is minimal for this technique, the throughput of the total uplink traffic in the wireless network does not change.

Table 4: Additional NAV duration contained in the modified ACK packets in different time period

Time (second)	4-5	5-6	6-7	7-8
NAV (μs)	500	1000	1500	2000
Time (second)	8-9	9-10	10-11	11-12
NAV (μs)	2500	3000	3500	4000

4. ANALYSIS

In this section, we analyze the evaluation results, and discuss the tradeoffs of the four mechanisms in detail. To compare and understand the evaluation results, we define *Effectiveness* and *Overhead* as the two metrics.

Effectiveness is defined as, for a protected station, in the application layer, the ratio between throughput of the uplink traffic in protected mode and the throughput of the uplink traffic in normal mode. Protected mode means that AP allocates medium resource to this uplink traffic by different mechanisms, while normal mode means that AP works normally without running any mechanism we propose.

$$Effectiveness = \frac{TH_{protected\_mode}}{TH_{normal\_mode}} \quad (1)$$

Figure 10 shows the *Effectiveness* of the protected uplink traffics of the evaluation results. This figure shows that in the evaluation results, *Dropping Packets*-based mechanism has the highest *Effectiveness*, while the *Delaying ACK*-based mechanism has the lowest *Effectiveness* over all the evaluation time. This results do not mean the *Dropping Packets*-based mechanism is the most effective mechanism.

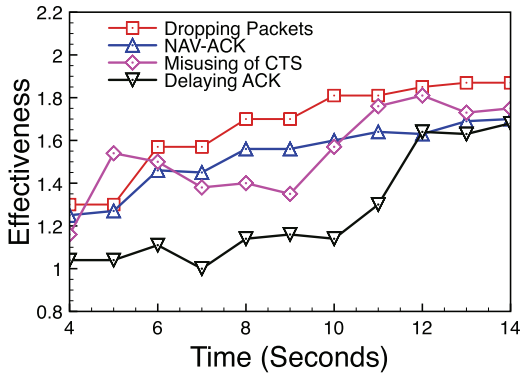

Figure 10: Effectiveness of the protected uplink traffic

AP cannot control the back-off time of the unprotected station in the *Dropping Packet*-based mechanism. If AP sets longer back-off time in the CTS frames or the *NAV-ACK* packet, the *CTS*-based mechanism or the *NAV-ACK*-based mechanism should also have higher *Effectiveness*.

Overhead is defined as, for a unprotected station, in the application layer, the amount of the delayed packets per second when AP allocates medium resources to protected stations for the different mechanisms. *Overhead* is computed as

$$Overhead = \frac{M_{delayed}}{T} \quad (2)$$

where $M_{delayed}$ is the amount of the delayed uplink packets in the unprotected station, and T is the length of the time period. Figure 11 shows the overhead of the unprotected uplink traffic of the evaluation results. The results show that both *Dropping Packets* and *Misusing of CTS*-based mechanisms causes large *Overhead* to the unprotected uplink traffic, while the *Overhead* caused by NAV-ACK-based mechanism is not so large. The overhead caused by *Delaying ACK*-based mechanism is small because the evaluation of *Delaying ACK*-based mechanism is conducted in a "slower" network, 802.11b network with maximum data rate of 2 Mbps.

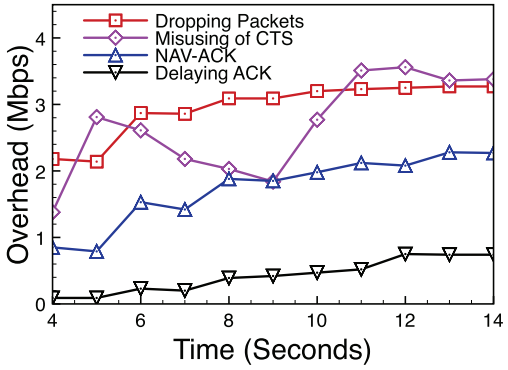

Figure 11: Overhead of the unprotected uplink traffic

Tradeoff of different mechanisms. Table 5 summarizes the four mechanisms in detail, and also compare them

Table 5: Mechanisms to control the uplink traffic in 802.11 wireless network

	Dropping Packet	Delaying ACK	Misusing of CTS	NAV-ACK	WMM
Modify AP	Yes	Yes	Yes	Yes	Yes
Modify Station	No	Yes	No	Yes	No
Additional Traffic	Yes	Yes	Yes	No	No
Block other stations	No	No	Yes	No	No
Collisions	No	No	Yes	No	No
Transparent to Application	Yes	Yes	Yes	Yes	No
Real Deployment	Yes	No	Yes	Yes	Yes

with WMM Extensions. Both *Dropping Packet*-based mechanism and *Misusing of CTS*-based mechanism are very simple, and only require minor modifications to the link layer of AP. However, *Delaying ACK*-based mechanism, and *NAV-ACK*-based mechanism require modifications on both wireless stations and AP (Please note that wireless stations do not accept the delayed ACK as a valid ACK packet, so we also need to modify the link layer of wireless station to deploy this mechanism). Additional traffic (control frames or retransmission) are caused in *Dropping Packet*, *Delaying ACK*, and *Misusing of CTS*-based mechanisms. The total throughput of the wireless network decreases because of the additional traffic caused by these mechanisms. However, the *NAV-ACK*-based mechanism causes no any additional traffics in the wireless network with high scalability. Also, *Delaying ACK*-based mechanism causes collisions as discussed in Section 2.2. All the four mechanisms we discuss in this paper control the uplink traffic in the link layer, which is transparent to application layer. *WMM Extensions* prioritizes the traffic flows based on different traffic classes. Applications tags the traffic flows with different priorities. However, if all the traffic flows come with the same level traffic class, WMM extension cannot control or prioritize the traffic flows.

Real Deployment. For real deployment, *Dropping Packet*, *Misusing of CTS*, and *NAV-ACK* are practical mechanisms. *Delaying ACK* is not suggested because it causes collisions. To achieve accurate control with high scalability, *NAV-ACK*-based mechanism are suggested, while *Misusing of CTS*-based mechanism is an effective solution if the traffic of a particular wireless station needs be protected. *Dropping Packet*-base mechanism has high *Effectiveness*, though this mechanism causes useless uplink traffics (retransmission).

5. RELATED WORK

Gkantsidis et al. propose a framework to manage the network traffic and resource in a small wired or wireless network [2]. In their work, the controller gathers information from all wired or wireless hosts in the small network, and jointly optimizes the resources. It requires to deploy special program on all hosts, to gather the application information on each host, and control the traffic of each host's applications, though it does not require any changes to the network devices. Carrera et al. [1] discuss scheduling the uplink traffic in a wireless home network and propose a selective silence mechanism by misusing the CTS frames. As we discussed, CTS-based mechanism protects the traffic bandwidth of protected stations effectively. However, it causes large amount of additional control traffic in the network, which decreases the available throughput of the wireless network. Also, scalability is another problem of the *misusing of CTS*-based mechanism. Yang et al. [3] propose a interactive visual tool to enable home owner to manage the home network. The visual tool designed by Yang et al. collects home network information by network tools, such as iptables, tc, hostapd, nmap, dnsmasq, and squid, and then shows the network condition to home owners.

6. CONCLUSION AND FUTURE WORK

In this paper, we discuss four different mechanisms to enable an access point to control the uplink traffic in IEEE 802.11 wireless network. We evaluate the four mechanisms using the ns3 simulator, and analyze the tradeoff of the different mechanisms. We also discuss the *Effectiveness* of different mechanisms, and *Overhead* caused by different mechanisms. The evaluation results show that the mechanisms we describe in this paper can provide granular control over uplink traffic flows. Finally, based on the evaluation results, we provide suggestions for the real deployment.

The mechanisms we propose and evaluate in this paper is the first step toward controlling the uplink traffic in 802.11 wireless network. As future work, we plan to evaluate the four mechanisms on the real wireless NICs and build deploy these in homes to better understand the practical challenges of controlling uplink flows in homes.

7. REFERENCES

[1] M. Carrera, P. Srikantha, M. May, and C. Rosenberg. SHAPE: Scheduling in wireless home network. Technical report, Technicolor and UPMC Paris Universitas and University of Waterloo, 2011.

[2] G. Gkantsidis, T. Karagiannis, P. Key, , B. Radunovic, E. Roftopoulos, and D. Manjunath. Traffic management and resource allocation in small wired/wireless network. In *The 6th International Conference on emerging Networking EXperiments and Technologies*, 2009.

[3] J. Yang, W. K. Edwards, and D. Haslem. Eden: Supporting home network management through interactive visula tools. In *24th Symposium on User Interface Software and Technology UIST*, 2010.