Abstract- IEEE 802.11-based Wireless LANs have become ubiquitous in coffee shops, office buildings, and hundreds of millions of residential homes. Access points, or APs, have been playing an important role in infrastructure Wireless LANs. An AP provides lots of services including cell identification, synchronization, authentication, distribution service, etc. Another important functionality of AP is to relay messages among wireless stations inside the same wireless cell. In current implementations, an AP needs to compete for the wireless channel with all other wireless stations using DCF protocol. Our objective in this paper is to design systematic and reproducible experiments to show that, with uncontrolled UDP traffic in the network, the AP becomes the system bottleneck and the system goodput could drop to an unacceptable level, mainly due to buffer overflow at the AP. E.g., in an 802.11g wireless network operating at 54Mbps, the saturation UDP goodput can be as low as only several Mbps, and TCP connections can be easily choked by UDP traffic for a long duration. We think this observation is important because UDP traffic volume is growing rapidly with the widely-deployed Voice over WiFi, wireless surveillance system, digital games, multimedia streaming applications, etc.

I. INTRODUCTION

A. Background and Motivation

IEEE 802.11 protocol [4] supports two kinds of operation modes: (1) in infrastructure mode, an Access Point (AP) acts as a central node. A station has to send its message to AP first, while the AP is responsible for forwarding all messages to the destinations, which can be either inside or outside of this wireless cell; (2) in ad hoc mode, there is no AP in the network and the wireless stations are able to communicate with each other directly. The infrastructure mode is the dominant mode widely adopted by most of the current wireless LANs, where AP performs the role of a portal (router) to Internet, as well as a central node of the local wireless cell.

The basic access method in the 802.11 MAC protocol is DCF (Distributed Coordination Function) based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) [4]. DCF employs a distributed CSMA/CA algorithm and an optional virtual carrier sense using RTS and CTS control frames. When using the DCF, before initiating a transmission, a station senses the channel to determine whether another station is transmitting. If the medium is found to be idle for an interval that exceeds the Distributed InterFrame Space (DIFS), the station proceeds with its transmission. However if the medium is busy, the transmission is deferred until the ongoing transmission terminates. A random interval, henceforth referred to as the backoff interval, is then selected; and used to initialize a backoff timer. The backoff timer is decremented as long as the channel is sensed idle, stopped when a transmission is detected on the channel, and reactivated when the channel is sensed idle again for more than a DIFS. The station transmits when the backoff timer reaches zero. When more than one node are counting down their backoff timers simultaneously, there is a probability that some of them have their timers reach zero at the same time slot, and start transmitting at the beginning of next time slot exactly at the same time, which results in a collision.

The performance of 802.11 DCF has been studied in the literature through analytical models, simulations, and experiments [8-17]. It is well known that DCF throughput degrades gracefully under increasing multiple access contention. However, previous study always assumes that AP is just a bridge between the local network and the outside network. As far as we know, there is no systematic study about the system goodput when intra-cell UDP traffic exists. A typical scenario with intensive intra-cell UDP traffic is shown in Fig. 1, which simulates an environment of a digital home employing IEEE 802.11 wireless LAN. In this scenario, lots of real time applications such as wireless surveillance system, voice over WiFi, HiFi over WiFi, as well as the Internet access service, are all supported by a wireless LAN with an AP.

![Figure 1. A typical scenario of digital home based on a wireless LAN](image-url)
Considering the performance of such a network, we have the following conjecture:

**Conjecture 1** Denote the saturation goodput of a single cell wireless LAN with $n$ senders as $T_s$, assuming that (1) DCF is employed; (2) the wireless channel is perfect and there is no transmission error; (3) the infrastructure mode has the same parameters as ad hoc mode. Consider a single cell wireless LAN with one AP and $n$ stations working under infrastructure mode. All the stations always have packets to be delivered to some destination which is another station in the same wireless LAN. We conjecture that the saturation UDP goodput of this system is $T_s/(1+n)$.

**Discussion** The above conjecture is based on the fact that DCF protocol tries to guarantee that every sender in the wireless LAN has the same chance to capture the medium. Given one AP and $n$ stations and at saturation condition, there are always $1+n$ senders, and hence the AP has only $1/(1+n)$ of chance to capture the medium. Goodput is defined as the number of data payload bits received by the destination per second. From a destination’s perspective, all its received packets must be sent from the AP; so the overall system goodput equals the number of data payload bits successfully sent out by the AP per second. Therefore we conclude that the saturation UDP goodput of the system is $T_s/(1+n)$.

In order to validate the above conjecture, we design a set of experiments to evaluate the performance of an 802.11g [5] single hop wireless LAN. The parameters of 802.11g are shown in Table I. The experiments are designed such that reproducible results could be easily obtained. We also study the following performance metrics: system goodput, packet loss rate, packet delay, at finite load. We finally study the downstream TCP performance with UDP background traffic.

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Chipset</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Thinkpad Z61M</td>
<td>Intel® PRO 3945 ABG</td>
</tr>
<tr>
<td>IBM Thinkpad T60</td>
<td>Intel® PRO 3945 ABG</td>
</tr>
<tr>
<td>IBM Thinkpad T43</td>
<td>Intel® PRO 2200BG</td>
</tr>
<tr>
<td>DELL Inspiron 700M</td>
<td>Intel® PRO 2200BG</td>
</tr>
<tr>
<td>DELL Inspiron 5100</td>
<td>Atheros</td>
</tr>
</tbody>
</table>

**TABLE I**

<table>
<thead>
<tr>
<th>IEEE 802.11G PHY CHARACTERISTICS [5]</th>
</tr>
</thead>
<tbody>
<tr>
<td>SlotTime</td>
</tr>
<tr>
<td>SIFSTime</td>
</tr>
<tr>
<td>DIFS</td>
</tr>
<tr>
<td>aCWmin</td>
</tr>
<tr>
<td>aCWmax</td>
</tr>
</tbody>
</table>

**B. Our Contributions**

We summarize our contributions as follows:

1. We propose a conjecture about the saturation UDP goodput for a wireless LAN with intra-cell UDP traffic, and we conduct reproducible experiments to prove the correctness of the conjecture.
2. We design a method to verify whether an 802.11 chipset conforms to the 802.11 DCF protocol or not.
3. We design a method to estimate the buffer size of an AP.
4. We evaluate the performance a wireless LAN with finite load of UDP traffic, including the goodput, average packet delay, and packet loss rate.
5. We show that downstream TCP sessions can easily be choked by intra-cell UDP traffic.

The remainder of the paper is organized as follows. In Section II, we introduce our experimental platform and tools. We all present the experiments of conformance test and also the experiments to estimate the AP’s buffer size. In Section III, we study the system UDP goodput at saturation condition. In Section IV, we study the system UDP goodput at finite load condition. We also study the packet delay and packet loss rate. In Section V, we study the impact of intra-cell UDP traffic on the performance of downstream TCP session. Section VI concludes the paper.

**II. EXPERIMENTAL PLATFORM AND METHOD**

**A. Experimental Platform and Tools**

We conduct experiments in both infrastructure mode and ad hoc mode. Since our focus is to study the problem caused by AP, we perform all the experiments in a near perfect indoor environment to minimize the noise effect. All the notebooks and AP are placed together on a roundtable. The configurations of these notebooks are shown in Table I. We used two different APs: one is D-Link DWL-2100AP, and another one is Linksys Wireless-G WRT54GL. All experiments have been repeated for five times, and the average results are reported in the paper.

All the notebooks are running windows XP with native 802.11 drivers from the vendor. To evaluate the saturation goodput, we developed our own UDP traffic generator which simply keeps on sending out UDP packets one by one (using blocking socket [6]). To evaluate non-saturation goodput, we choose to use iperf [3] as the traffic generator because it can provide detailed report on the packet loss rate.

**TABLE II**

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Chipset</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Thinkpad Z61M</td>
<td>Intel® PRO 3945 ABG</td>
</tr>
<tr>
<td>IBM Thinkpad T60</td>
<td>Intel® PRO 3945 ABG</td>
</tr>
<tr>
<td>IBM Thinkpad T43</td>
<td>Intel® PRO 2200BG</td>
</tr>
<tr>
<td>DELL Inspiron 700M</td>
<td>Intel® PRO 2200BG</td>
</tr>
<tr>
<td>DELL Inspiron 5100</td>
<td>Atheros</td>
</tr>
</tbody>
</table>
We use a dedicated notebook DELL ISPRION 5100, installed with a commercial software AiroPeek NX [1], to capture the wireless packets in the air. It can capture all the wireless frames including data, management, and control frames. Another sniffer software Ethereal [2] is used for measuring the TCP performance.

B. Conformance Test

To design reproducible experiments, it is important to use devices which strictly conform to 802.11 DCF protocol. We design two experiments for testing whether a wireless device conforms to the 802.11 DCF:

[Experiment 1] Let one tested wireless device be the only sender in the wireless LAN, and this wireless device creates saturated UDP traffic. According to the 802.11 DCF protocol, the number of empty time slots between two successful frame transmissions shall be uniformly distributed within range [0, \(aCW\min\)-1]. If the observed distribution does not follow a uniform distribution, we can conclude that the tested device violates the 802.11 DCF protocol. If the device passes this test, we continue to carry out the second experiment. Through this experiment, we can also get the value of \(aCW\min\). In all the notebooks we tested, the value of \(aCW\min\) for 802.11g is 16.

[Experiment 2] Let two tested wireless devices be the only two senders in the wireless LAN, and both of them create saturated UDP traffic. We have derived an analytical model to calculate the distribution of the number of empty time slots between two successful transmissions [18]. By comparing the observed distribution with the one from our analytical model, we can know whether the tested wireless devices conform to the 802.11 DCF protocol or not. Fig. 2 shows the distribution from our analytical model and our experimental results.

We do find that some other PCI desktop wireless cards (such as D-link DWL-G510) do not strictly comply with 802.11g. When they are used together with the devices listed in Table one, most of the bandwidth is grabbed by the PCI desktop wireless cards. These cards failed to pass our conformance test, and hence we do not use these PCI desktop wireless cards in our experiments.

C. Estimation of AP’s buffer size

The buffer size of an AP is usually unknown to the users. In this subsection, we present a novel method to estimate the buffer size of an AP, which is based on the following observation: in the infrastructure mode, if there are two stations keep on sending packets to each other through the AP, the chance that AP seizes the channel is roughly 1/3, while the two stations get the remaining 2/3 bandwidth. Hence, the AP’s buffer will be full after a short time. At the moment when the buffer is full, the AP starts to drop packets.

We design the following experiment to estimate the buffer size of an AP, illustrated in Fig. 3: at the beginning, the AP’s outgoing buffer is empty. Then we let STA X and STA Y start to send out UDP packets to each other as fast as possible. All the packets are actually delivered to the AP, and it is the AP’s responsibility to forward the packets to the destination. The packets sent out by STA X are identified sequentially as X(1), X(2), X(3), etc; and the packets sent out by STA Y are identified as Y(1), Y(2), Y(3), etc. Without loss of generality, we assume the first observed lost packet is X(\(N\)). This simply means that all packets X(i), \(i < N\), have been received by STA Y, and X(\(N\)) is not received by STA Y. Denote the last packet sent out by STA Y before X(\(N\)) as Y(\(M\)). Then before the packet loss happens, there are totally \(N-1+M\) packets sent out by STA X and STA Y. These packets are either successfully delivered by the AP, or are stored in the AP’s buffer. The number of packets delivered by the AP, denoted by A, can be easily obtained from the sniffer. Then the AP’s buffer size can be calculated as \(N-I+M-A\).

We used the above method to estimate the buffer size of two different APs. For each AP, we repeated the experiments several times and the results are almost the same, which verifies the correctness of our method. We also conduct the experiments for different UDP packet size (from 100 bytes to 1472 bytes), and it turns out that the APs can only accommodate a fixed number of packets which is independent of the packet size. For the D-Link DWL-2100AP AP, the
buffer size is 368 packets. For the Linksys Wireless-G WRT54GL AP, the buffer size is 106 packets.

III. MEASUREMENT UNDER SATURATION CONDITION

In this section, we study the system goodput under saturation conditions, i.e., the transmission queue of each station is always nonempty. The system goodput is defined as the total number of UDP payload bits correctly received by the destinations per second. Due to the limited space, we only present the results of an 802.11g WLAN. We conduct experiments to test the saturation goodput of a single cell WLAN operating at ad hoc mode, and infrastructure mode, respectively. All the notebooks generate saturated UDP traffic to another notebook in the same WLAN. In all our experiments, RTC/CTS scheme is disabled.

Fig. 4 shows the saturation goodput for an 802.11g WLAN operating at ad hoc mode and infrastructure mode, respectively, both with a UDP payload of 1472 bytes. When operating at ad hoc mode, we can observe that the system goodput can achieve the maximum with two or three senders. It is higher than the goodput of one sender due to the concurrent countdown effect. But once the number of senders is larger than 3, the collision probability grows and therefore the system goodput degrades gradually. The bottom curve shows the system goodput under infrastructure mode. It is obvious that the system goodput drops seriously with the increase of number of senders. In fact, there are two main reasons for this: (1) in order to send a packet from the source to the destination, the same packet has to be transmitted twice; (2) because the AP has the same priority as normal wireless stations, the AP has roughly \( 1/(1+n) \) of chance to deliver a packet. Without any rate control, the senders’ traffic will easily fill up AP’s buffer, and a large portion of the packets will be dropped by the AP, therefore wasting lots of bandwidth.

![Figure 4. Saturation goodput versus number of senders, UDP payload = 1472 bytes](image)

Our experimental results validate the main argument of Conjecture 1 presented in Section I. But Conjecture 1 is not precise because assumption 3 made in the conjecture is not true in reality. In 802.11g infrastructure mode, the backoff time slot equals \( 9\mu s \); while in 802.11g ad hoc mode, the backoff time slot equals \( 20\mu s \). Thus the bandwidth utilization of ad hoc mode is lower than that of infrastructure mode. Another important factor which affects the system goodput is the dynamic rate shifting algorithm implemented in the wireless chips. After two consecutive failures, the transmitter will decrease the physical transmission rate, and hence the measured system goodput is lower than the theoretical bound. This phenomenon has also been reported in [15].

IV. MEASUREMENT UNDER FINITE LOAD CONDITION

In this section, we study the system performance under finite UDP traffic load. We will measure the system goodput, average packet delay, and packet loss rate. The experiments are conducted on an 802.11g WLAN with 4 notebooks and one AP, working at infrastructure mode.

A. System Goodput

From Fig. 5, we can see that under a light traffic load (i.e., less than 3Mbps per node), the system goodput grows linearly with the increase of offered load. The “offered load per node” is the parameter given to the traffic generator iperf. In the range of 0-3Mbps, the aggregated offered load is less than 12Mbps, and the AP can use the residual bandwidth to forward all offered traffic and therefore the system goodput equals the aggregated offered load. Once the aggregated offered load is larger than half of the system bandwidth, the AP will not be able to forward all the incoming packets in time. AP’s buffer will grow, and finally incoming packets will be dropped once the buffer becomes full. Once the offered load per node is beyond a threshold, the whole system will be saturated, i.e., all the notebooks and the AP are competing for the wireless medium all the time. In this case, the real traffic sent out by each notebook will become a constant which is less than the offered load. And the system goodput also becomes a constant value which depends on the capability of the AP in competing for the wireless medium.

B. Average Packet Delay

Fig. 6 shows the average packet delay for those successfully received packets. Under light traffic load (say, less than 3Mbps per node), there is almost no queuing delay in the AP, and hence the average packet delay is close to twice of the frame transmission time due to the low collision probability. Once the traffic load is greater than the AP’s bandwidth, queuing delay will dominate the end-to-end packet delay. As we can see from Fig. 6, the buffer size has an impact on the packet delay when the traffic is heavy: a large buffer size can accommodate more packets at the expense of a longer queuing delay. When the network is heavy, we can see that the average delay is linear to the buffer size: this is because the buffer is always full and the queuing delay dominates the end-to-end delay.
C. Packet Drop Rate

The packet drop rates are shown in Fig. 7. At light traffic load, the packet drop rate is close to 0. At heavy traffic load, the packet drop rate grows almost linearly with the increase of offered load. In the range of 3.5-4Mbps of offered load, the system is heavily loaded but the AP’s buffer is not full. In such a case, we can observe a lightly smaller packet drop rate for the AP with a bigger buffer. But as shown in Fig. 6, this advantage of lower packet drop rate is at the expense of a longer packet delay. For real time applications, an AP with a smaller buffer size is preferred.

V. Measurement Of Downstream TCP Goodput

In this section, we present experimental results of downstream TCP goodput when background intra-cell UDP traffic exists. The purpose is to show that a small volume of intra-cell UDP traffic can have a great impact on the TCP performance, mainly due to the buffer overflow at the AP. In our experiments, one notebook acts as a TCP client where a PC acts as a TCP server which is directly connected to the AP by Ethernet. Some other notebooks generate intra-cell UDP traffic.

In the first set of experiments, we gradually increase the data rate of UDP traffic and then watch the corresponding TCP goodput. Fig. 8 shows the results for the experiments in which packet loss is not observed at the AP. We can see that the TCP goodput drops almost linearly with the increase of UDP traffic load.

In the second set of experiments, we first start a single TCP session. After 10 seconds, we start 3 UDP sessions, each with a constant data rate of 4.3Mbps, to investigate the impact on the TCP goodput. After another 100 seconds, we stop the UDP sessions. The TCP goodput is shown in Fig. 9. At the beginning, the TCP goodput can be as high as 22Mbps. Once the UDP sessions are created, the TCP goodput suddenly drops to the order of 1Mbps. Under this traffic configuration, packets start to accumulate at the AP’s buffer. At time 70th second, we observe that some packets are dropped by the AP, and the TCP goodput drops to zero because of the loss of several consecutive TCP ACK segments.
In the third sets of experiments, we study the impact of small UDP packets on the TCP goodput. We use 5 UDP senders which generate UDP packets with a payload of 100 bytes. The results are shown in Fig. 10. We can see that, with a very light UDP traffic load (no more than 0.5Mbps per node), the TCP goodput drops to zero.

VI. CONCLUSION AND FUTURE WORK

We first designed experiments for conformance test on 802.11 chipsets, as we found some products are using proprietary methods which do not fully comply with the 802.11 standard. We then presented a method to estimate the buffer size of an AP, which has an impact on the average packet delay. Next, we conducted experiments to measure the UDP saturation goodput. We also compared the results to ad hoc mode where there is no AP in the network. Finally, we designed experiments to evaluate the impact of UDP traffic on the TCP performance. Our main conclusion is that, the current implementation of AP based on 802.11 DCF cannot survive in an environment with considerable volume of UDP traffic. Our future work includes the investigation and evaluation of different schemes to solve the AP’s buffer overflow problem.

ACKNOWLEDGEMENT

This work was partially supported by Hong Kong Baptist University under grant FRG/06-07/II-69 and Hong Kong RGC Central Allocation grant HKBU 1/05C.

REFERENCES