

12 How many downloads can a Wi-Fi access point simultaneously handle?

12.1 Motivation

Wi-Fi has become the system of choice for access to Internet services inside buildings, offices, malls, airports, etc. In order to obtain access to the Internet over Wi-Fi a user connects his/her mobile device (a laptop or a cellphone, for example) to a nearby Wi-Fi access point (AP). A popular use of such a connection is to download a document, or a music file; in such an application, the user's desire is to download the file as quickly as possible, i.e., to get a high throughput during the download. It is a common experience that as the number of users connected to an AP increases, the throughput obtained by all the users decreases, thereby increasing the time taken to download their files. The following question can be asked in this context.

If during the download, a user expects to get a throughput of at least θ bytes per second, what is the maximum number of users (say, n_θ) up to which the throughput obtained by every user is at least θ . We can say that n_θ is the *capacity* of this simple Wi-Fi network for the *Quality of Service (QoS)* objective θ .¹

12.2 Objective

In this experiment we will learn how to obtain n_θ in a simple WiFi network where the packet loss due to channel errors is 0. In this process we will understand some interesting facts about how WiFi networks perform when doing file transfers.

12.3 Theory

In NetSim, we will set up a network comprising a server that carries a large number of large files that the users would like to download into their mobile devices. The server is connected to a Wi-Fi AP, with the IEEE 802.11b version of the protocol, via an Ethernet switch. Several mobile devices (say, N) are associated with the AP, each downloading one of the files in the server. The Ethernet speed is 100Mbps, whereas the mobile devices are connected to the AP at 11Mbps, which is one of the IEEE 802.11b speeds.

¹ It may be noted that the term *capacity* has several connotations in communications. Our use of the word here must not be confused with the notion of *information theoretic capacity* of a communication channel.

We observe, from the above description, that the file transfer throughputs will be limited by the wireless links between the AP and the mobile devices (since the Ethernet speed is much larger than the Wi-Fi channel speed). There are two interacting mechanisms that will govern the throughputs that the individual users will get:

1. The Wi-Fi medium access control (MAC) determines how the mobile devices obtain access to the wireless medium. There is one instance of the WiFi MAC at each of the mobile devices.
2. The end-to-end protocol, TCP, controls the sharing of the wireless bandwidth between the ongoing file transfers. In our experiment, there will be one instance of TCP between the server and each of the mobile devices.

For simplicity, the default implementation of TCP in NetSim does not implement the delayed ACK mechanism. This implies that a TCP receiver returns an ACK for every received packet. In the system that we are simulating, the server is the transmitter for all the TCP connections, and each user's mobile device is the corresponding receiver.

Suppose each of the N TCP connection transmits one packet to its corresponding mobile device; then each mobile device will have to return an ACK. For this to happen, the AP must send N packets, and each of the N mobile devices must send back N ACKs. Thus, for the file transfers to progress, the AP needs to N packets for each packet (i.e., ACK) returned by each mobile device. We conclude that, in steady state, the AP must send as many packets as all the mobile devices send, thus requiring equal channel access to the AP as to all the mobile devices together.

At this point, it is important to recall that when several nodes (say, an AP and associated mobile devices) contend for the channel, the WiFi medium access control provides fair access at the packet level, i.e., each contending device has an equal chance of succeeding in transmitting a packet over the channel. Now consider the system that we have set up in this present experiment. There are N mobile devices associated with one AP. Suppose, for example, 10 of them ($N \geq 10$) all have a packet to transmit (and none other has a packet). By the fair access property of the WiFi MAC, each of these 10 nodes, along with the AP, has an equal probability of successfully transmitting. It follows, by the packet level fair access property, that each node will have a probability of $\frac{1}{11}$ of succeeding in transmitting its packet. If this situation continues, the channel access ratio to the AP will be inadequate and the equal channel access argued in the previous paragraph will be violated. It follows from this that, on the average, roughly only one mobile device will have an ACK packet in it; the AP will contend with one other node, thus getting half the packet transmission opportunities.

With the just two nodes contending, the collision probability is small (~ 0.06) and the probability of packet discard is negligibly small. Thus, the TCP window for every transfer will grow to the maximum window size. The entire window worth of TCP data packets for the N sessions will be in the AP buffer, except for a very small number of packets (averaging to about 1) which will appear as ACKs in the mobile devices.

It follows that, in steady state, the system will look like two contending WiFi nodes, one with TCP data packets and the other with TCP ACK packets. This will be the case no matter how many downloading mobile devices there are. The total throughput can be obtained by setting up the model of two saturated nodes, one with TCP data packets, and the other with TCP ACK packets. The data packets of all the TCP connections will be randomly ordered in the AP buffer, so that the head-of-the-line packet will belong to any particular mobile device with probability $\frac{1}{N}$. This throughput is shared equally between the N mobile devices.

Now suppose that the TCP data packet throughput with the two-node model is θ . Then

$$n_{\theta} = \left\lfloor \frac{\theta}{\theta} \right\rfloor$$

where the $\lfloor x \rfloor$ denotes the largest integer less than or equal to x . Use NetSim to verify that for an 11Mbps Wi-Fi speed, with RTS/CTS enabled the total TCP throughput is 3.4 Mbps. If $\theta = 0.65 \text{ Mbps}$, then $n_{\theta} = \left\lfloor \frac{3.8}{0.65} \right\rfloor = 5$. In this example, if $N = 5$ the download throughput obtained by each of them will be 0.68 Mbps , but if one more downloading device is added then each will get a throughput less than $\theta = 0.65 \text{ Mbps}$. We say that the capacity of this network for a target throughput of 0.65 Mbps is 5.

12.4 Procedure

Open NetSim and click **Examples > Experiments > How-many-downloads-can-a-Wi-Fi-access-point-simultaneously-handle? > Sample-1** as shown below **Figure 12-1**.

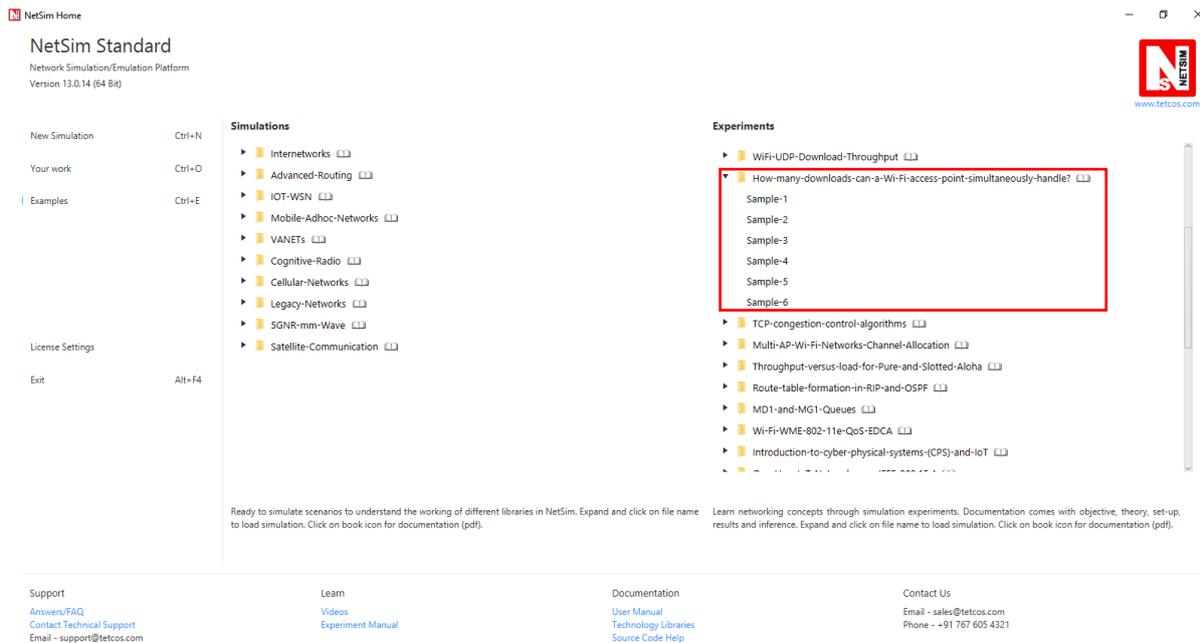


Figure 12-1: Experiments List

NetSim UI displays the configuration file corresponding to this experiment **Figure 12-2**.

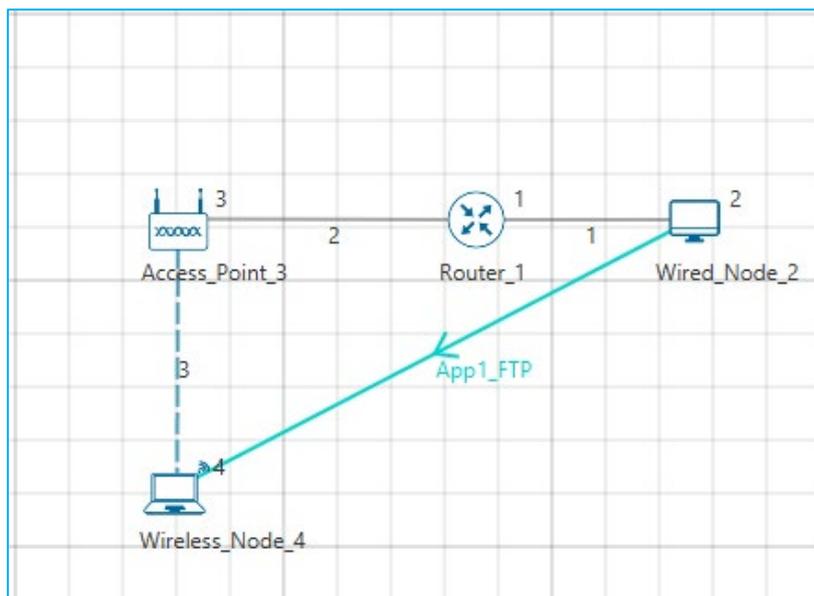


Figure 12-2: WiFi Network with Single TCP Download

12.5 Procedure

The following set of procedures were done to generate this sample.

Step 1: A network scenario is designed in the NetSim GUI comprising of 1 Wired Node, 1 Wireless Node, 1 Access Point, and 1 Router in the “**Internetworks**” Network Library.

Step 2: In the Interface (WIRELESS) > Data Link Layer Properties of Wireless Node 4 and Access point, Short Retry Limit was set to 7, Long Retry Limit was set to 4 and RTS Threshold was set to 1000 bytes. Medium-Access-Protocol is set to DCF **Figure 12-3**.

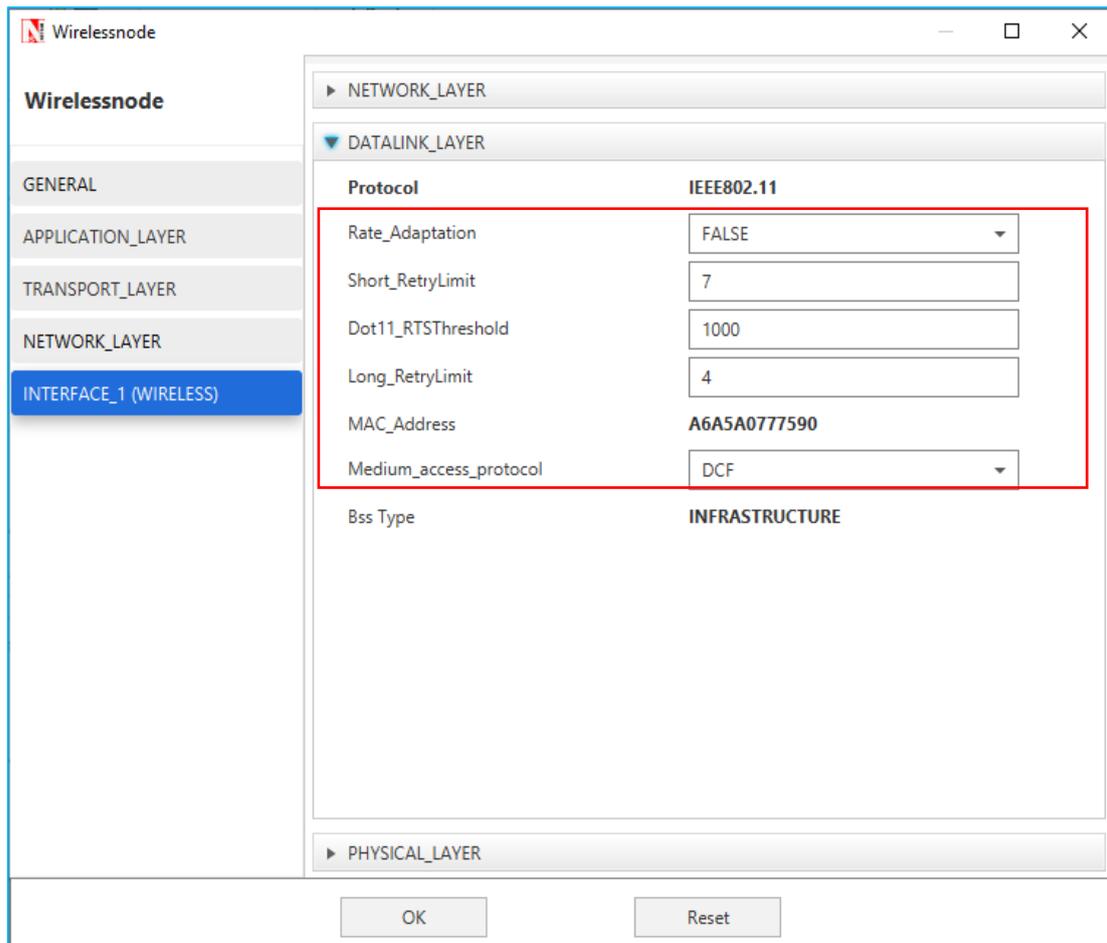


Figure 12-3: Data Link Layer Properties

Step 3: Right-click the link ID (of a wired link) and select Properties to access the link's properties. The Link Properties are set according to the values given in the below **Table 12-1**.

Wired Link	
Max Uplink Speed (Mbps)	100
Max Downlink Speed (Mbps)	100
Uplink BER	0
Downlink BER	0
Uplink Propagation Delay (μ s)	0
Downlink Propagation Delay (μ s)	0

Wireless Link	
Channel Characteristics	No path loss

Table 12-1: Detailed Wired/Wireless Links Properties

Step 4: Right click on **App1 FTP** and select Properties or click on the Application icon present in the top ribbon/toolbar.

An FTP Application is generated from Wired Node 2 i.e., Source to Wireless Node 4 i.e., Destination with File Size set to 10,000,000 Bytes and Inter Arrival Time set to 20 s.

Step 5: Enable the plots, run the Simulation for 15 Seconds, and note down the throughput.

Sample 2:

The following changes in settings are done from the previous sample:

Step 1: The number of Wireless Nodes is increased to 5 and FTP applications are generated from Wired Node 2 to each of the Wireless Nodes as shown below **Figure 12-4**.

No. of wireless nodes = 5

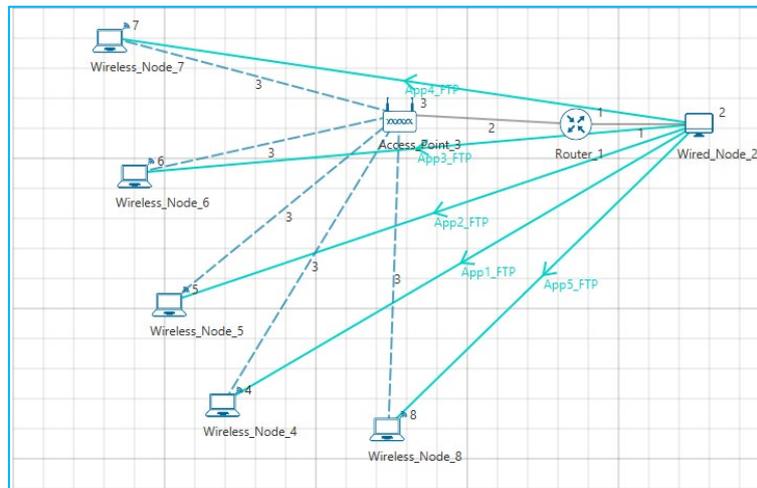


Figure 12-4: WiFi Network with Multiple TCP Download

Application Properties

Properties	App1	App2	App3	App4	App5
Application Type	FTP	FTP	FTP	FTP	FTP
Source Id	2	2	2	2	2
Destination Id	4	5	6	7	8
File size (Bytes)	10,000,000	10,000,000	10,000,000	10,000,000	10,000,000
File Inter arrival time	20 s				

Table 12-2: Detailed Application properties

Step 2: Enable the plots, run the Simulation for 15 Seconds, and note down the throughput.

NOTE: Follow the same procedure for next samples with wireless nodes 10, 15, 20, 25 and note down the sum of throughputs for all applications.

12.6 Measurements and Output

Aggregated download throughput with different values of N (wireless nodes) is shown below

Table 12-3.

$$\text{Throughput Per Device (Mbps)} = \frac{\text{Sum of throughputs (Mbps)}}{\text{Number of Devices}}$$

Sample Number	Number of Devices	Sum of throughputs (Mbps)	Throughput Per Device (Mbps)
1	1	3.39	3.39
2	5	3.40	0.68
3	10	3.18	0.32
4	15	3.23	0.21
5	20	3.24	0.16
6	25	3.26	0.13

Table 12-3: Aggregated download throughput for different number of wireless nodes

Plot

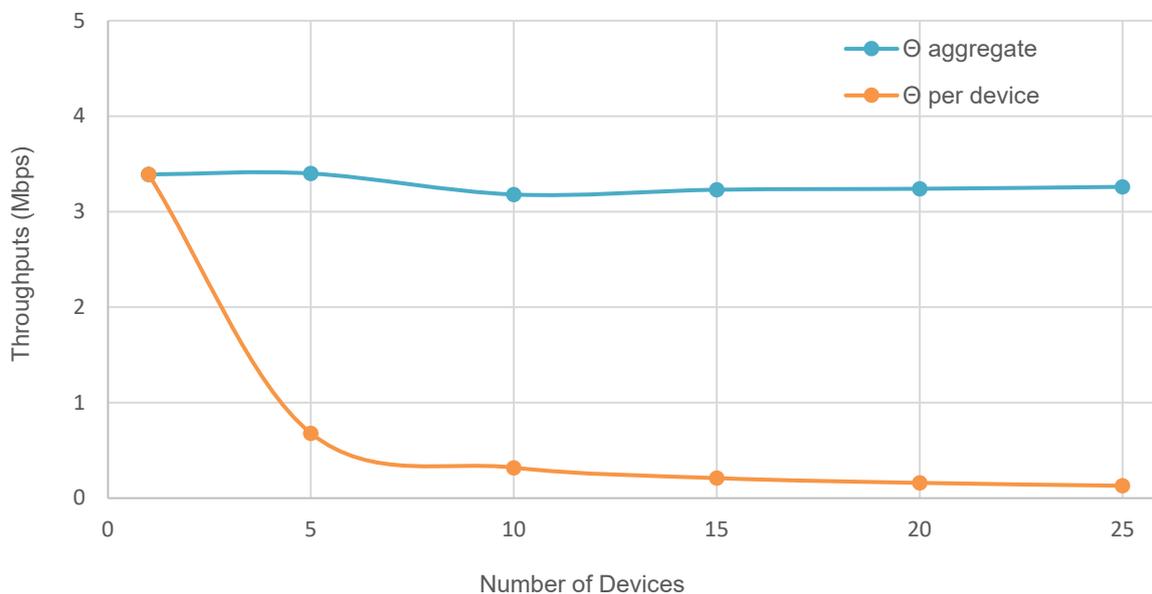


Figure 12-5: Plot of Number of devices Vs. Throughputs (Mbps)

NOTE: In the referred paper we see that, a throughput value for 11 Mbps WLAN is 3.8 Mbps. Please note that this is the aggregate PHY throughput of the AP. However, in NetSim, we are calculating the total Application throughput.

To derive the PHY layer throughput from the APP layer throughput, we need to add overheads of all layers **Table 12-4**.

Layer	Overhead (Bytes)
Transport Layer	20
Network Layer	20
MAC Layer	40
PHY layer	$48\mu\text{s} = (11*48)/8 = 66$
Total Overhead	146

Table 12-4: Overhead of different layers

$$PHY_Throughput = APP_Throughput * 1606/1460 = 3.42*1606/1460 = 3.76 \text{ Mbps}$$

12.7 Inference

We see that as the number of devices increase the aggregate (combined) throughput remains constant whereas the throughput per user decreases.

As discussed earlier, our goal was to identify that if during the download, a user expects to get a throughput of at least θ bytes per second, what is the maximum number of users (say, n_θ)?

If we set θ to be 650 Kbps, then we see that from the output table that the maximum number of users who can simultaneously download files is 5 (n_θ)

12.8 Reference Documents

1. *Analytical models for capacity estimation of IEEE 802.11 WLANs using DCF for internet applications.* George Kuriakose, Sri Harsha, Anurag Kumar, Vinod Sharma