

# 11 WiFi: UDP Download Throughput

## 11.1 The Setup and Motivation

The most basic packet transfer service offered by the Internet is called the “datagram” service, in which a series of packets are transmitted to a receiver without any packet loss recovery, flow control, or congestion control. The Internet’s UDP protocol implements the datagram service. In this experiment we will study the performance of UDP transfers from a server on a wireline local area network to WiFi Stations (STA), via WiFi Access Points (AP). The schematic of the network that we will be simulating in NetSim is shown in the figure below **Figure 11-1**.

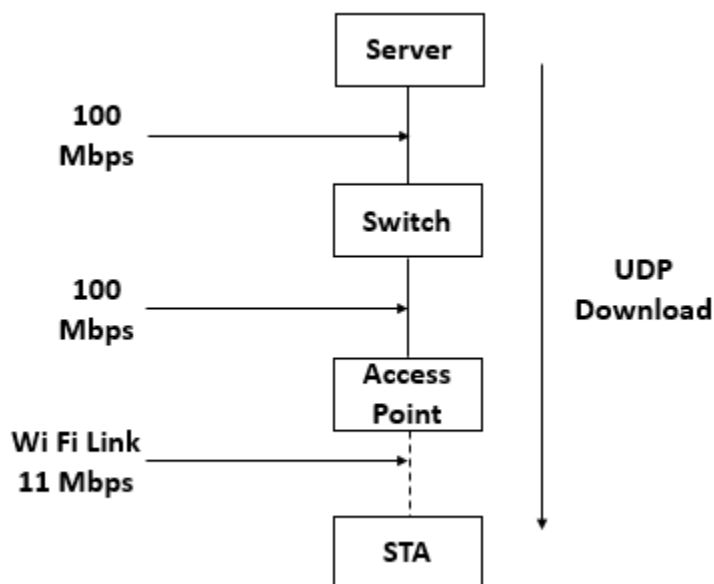


Figure 11-1: Network topology and application flow from server to STA via a single AP

The server, which contains the data that needs to be transferred to the STAs (say, laptops), is connected by a 100 Mbps switched Ethernet link to an Ethernet switch, which is, in turn, connected to the WiFi APs. Each AP is associated (i.e., connected) at 11 Mbps to a single STA. The objective is to transfer a large number of packets (say, constituting a video) from the server to each of the STAs, the packet stream to each of the STAs being different (e.g., each STA is receiving a different video from the server). In this experiment, we are interested in studying the limitation that the WiFi link places on the data transfers. We assume that the server transmits the packets at a high enough rate so that the queues at the APs fill up, and the rate of the UDP transfers is, therefore, governed by the WiFi link. It may be noted that, in practice, there will be a

flow control mechanism between each STA and the server, that will control the rate at which the server releases packets, in order to prevent buffer overflow at the APs.

In this setting, this experiment will ask one precise question. With the buffers at the AP full, at what rate will the WiFi protocol transfer the packets from the APs to the STAs over the wireless link. We will study two cases:

1. A single AP and a single STA: Since there is only one transmitter in this wireless network (namely, the AP), there is no contention, and the rate of packet transfer over the link will be governed by the basic overheads in the protocol, such as the interframe spacings, packet header overheads, transmit-receive turn-around times, and acknowledgement times. We will begin by a simple calculation (essentially timing book-keeping) that will predict the UDP throughput, and then we will verify our calculation using the NetSim simulator.
2. Multiple APs and one STA for each AP: This is the more common situation (for example neighboring apartments in a building, each with one AP and one laptop, all drawing data from the Internet service provider). The performance of such a system depends on the wireless propagation path-loss between the various APs. A predictive analysis is difficult in the general case. For deriving some insight, we will study the case where all the APs are close to each other, and thus exactly one transmission from AP to an STA can be successful at any time. If two or more APs transmit together, then all the transmissions are not successful. Even in this case, the analysis mathematically complex and is available in, Anurag Kumar, D. Manjunath and Joy Kuri. 2008: Wireless Networking. Sec 7.4

## 11.2 Predicting the UDP Throughput

### 11.2.1 One AP and one STA

As stated above, in the setup described, the AP queue is full. Thus, after a packet is completely transmitted over the wireless link, immediately the process for transmitting the next packet starts. This is illustrated by the upper part of the figure below, where the successive packets from the AP are shown as being sent back-to-back. The time taken to send a packet is, however, not just the time to clock out the physical bits corresponding to the packet over the Wi-Fi medium. After the completion of a packet transfer, the AP's Wi-Fi transmitter waits for a Distributed Coordination Function Inter-Frame Space (DIFS), followed by a backoff that is chosen randomly between 1 and 32 slots. Upon the completion of the backoff, the packet transmission starts. Each packet carries physical layer overheads, MAC layer overheads, and

IP overheads. After the transmission of the packet, there is a Short Inter-Frame Space (SIFS), which gives time to the receiver (namely, the STA) to transition from the listening mode to the transmit mode. After the SIFS, the STA sends back a MAC acknowledgement (ACK). This completes the transmission of one UDP packet from the AP to the STA. Immediately, the process for sending the next packet can start. The details of the various timings involved in sending a single UDP packet are shown in the lower part of the figure below **Figure 11-2**.

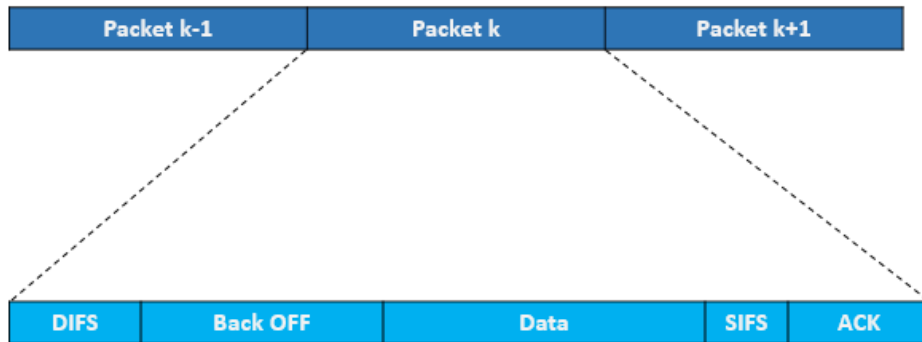


Figure 11-2: Detailed view of transmission of a single packet in WiFi involving single AP-STA. In this experiment, the payload in each packet is the same (1450 Bytes). Since the packets are sent back-to-back, and the state of the system is identical at the beginning of each packet transmission, the throughput (in Mbps) is computed by the following simple (and intuitive) relation.

### 11.2.1.1 Without RTS / CTS

$$UDP\ Throughput\ (Mbps) = \frac{Application\ Payload\ in\ Packet\ (bits)}{Average\ Time\ per\ Packet(\mu s)}$$

$$\begin{aligned} Average\ time\ per\ packet\ (\mu s) \\ &= DIFS + Average\ Backoff\ time + Packet\ Transmission\ Time + SIFS \\ &+ Ack\ Transmission\ Time \end{aligned}$$

$$Packet\ Transmission\ Time\ (\mu s) = Preamble\ time + (MPDU\ Size/Data\ rate)$$

$$Average\ Backoff\ time\ (\mu s) = (CW_{min}/2) * Slot\ Time$$

$$Ack\ Transmission\ Time\ (\mu s) = Preamble\ time + (Ack\ Packet\ size/Ack\ data\ rate)$$

$$DIFS\ (\mu s) = SIFS + 2 * Slot\ Time$$

$$Average\ Backoff\ time\ (\mu s) = (CW_{min}/2) * Slot\ Time$$

where

$$SIFS = 10 \mu s \text{ and Slot Time} = 20 \mu s$$

$$CW_{min} = 31 \text{ slots for } 802.11b$$

$$DIFS = SIFS + 2 \times \text{Slot Time} = 10 \mu s + 2 \times 20 \mu s = 50 \mu s$$

$$\text{Average Backoff Time} = 310 \mu s$$

$$\text{Packet Transmission Time} = 192 \mu s + \frac{1518 \times 8 \text{ bits}}{11 \text{ Mbps}} = 1296 \mu s$$

$$\text{Preamble time} = 192 \mu s \text{ for } 802.11b$$

$$\text{MPDU Size} = 1450 + 8 + 20 + 40 = 1518 \text{ Bytes}$$

$$\text{Ack Transmission Time} = 192 \mu s + \frac{14 \text{ Bytes} * 8}{1 \text{ Mbps}} = 304 \mu s$$

$$\text{Application Payload} = 1450 \text{ Bytes}$$

$$\text{Average time per packet} = 50 + 310 + 1296 + 10 + 304 = 1970 \mu s$$

$$\text{UDP Throughput} = \frac{1450 \times 8}{1970} = 5.89 \text{ Mbps}$$

### 11.2.1.2 With RTS/CTS

$$\text{UDP Throughput (Mbps)} = \frac{\text{Application Payload in Packet (bits)}}{\text{Average Time per Packet}(\mu s)}$$

$$\text{Average time per packet } (\mu s)$$

$$\begin{aligned} &= DIFS + \text{RTS Packet Transmission Time} + \text{CTS Packet Transmission Time} \\ &+ \text{Average Backoff time} + \text{Packet Transmission Time} + SIFS \\ &+ \text{Ack Transmission Time} \end{aligned}$$

$$\begin{aligned} \text{RTS packet transmission time} &= \text{Preamble time} + \left( \frac{\text{RTS Packet payload}}{\text{Data Rate}} \right) = 192 + 20 \times \left( \frac{8}{1} \right) \\ &= 352 \mu s \end{aligned}$$

$$\begin{aligned} \text{CTS packet transmission time} &= \text{Preamble time} + \left( \frac{\text{CTS Packet payload}}{\text{Data Rate}} \right) = 192 + 14 \times \left( \frac{8}{1} \right) \\ &= 304 \mu s \end{aligned}$$

$Average\ time\ per\ packet = 50 + 352 + 304 + 310 + 1296 + 10 + 304 = 2626\ \mu s$

$$UDP\ throughput = \frac{1460 \times 8}{2626} = 4.45\ Mbps$$

### 11.2.2 Multiple APs (near each other) and one STA per AP

Since the AP queues are full, on the WiFi medium the packet transmission can still be viewed as being back-to-back as shown in the upper part of the figure below

**Figure 11-3.** However, since there are multiple contending AP-STA links, there are two differences between this figure and the one shown above (for the single AP and single STA case **Figure 11-2**).

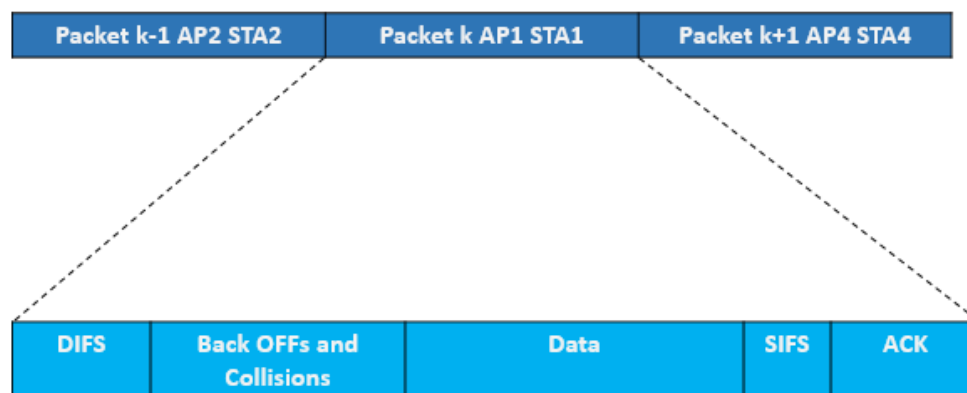


Figure 11-3: Detailed view of transmission of a single packet in WiFi involving Multiple AP-STA

- Within each transmission period, there is now a “backoffs and collisions” period, where in the figure above we only showed a “backoff” period. Access to the channel is by contention, collision, and backoff, and this “backoffs and collisions” duration is the time taken to select one transmitting AP.
- The other difference is that, after each “backoffs and collisions” period, any one AP-STA pair “wins” the contention, and the corresponding AP can then send a packet. It turns out that the contention mechanism is such that each of the AP-STA pairs can succeed with equal probability, independent of the pair that has previously been successful. Thus, if there are, say, 5 AP-STA pairs, then each successful packet transmission will be from any of these pairs with a probability of 0.2.

With reference to the figure above, note that, all the APs are contending to send their packets to their respective STAs, and the “Backoffs and Collisions” time is due to all the APs. However, finally, only one packet transmission succeeds. We will attribute all the contention overheads to the successful transmission of this packet. Thus, we will call the time duration from the beginning of a DIFS until the end of the ACK for the transmitted packet as the “effective” time

taken to transmit that packet on the wireless medium. The average of these effective packet transmission times can be called the “Average time per Packet.”

With this discussion, and the upper part of the figure above, it follows that the following expression still holds.

$$Total\ UDP\ Throughput\ (Mbps) = \frac{Application\ Payload\ in\ Packet\ (bits)}{Average\ Time\ per\ Packet(\mu s)}$$

We observe from the figure that the average time per packet will be larger than when there is a single AP-STA pair. Hence, the total UDP throughput will be smaller when there are multiple AP-STA pairs (since the “Application Payload in the Packet” is the same in both cases.

Having obtained the total throughput over all the AP-STA pairs in this manner, by the fact that each packet transmission is with equal probability from any of the AP-STA pairs, the UDP throughput for each AP-STA pair (for  $N$  pairs) is just  $\frac{1}{N}$  of the total throughput.

### 11.3 Network Setup

Open NetSim and click **Examples > Experiments > WiFi-UDP-Download-Throughput > Part-1 > Without-RTS-CTS** as shown below **Figure 11-4**.

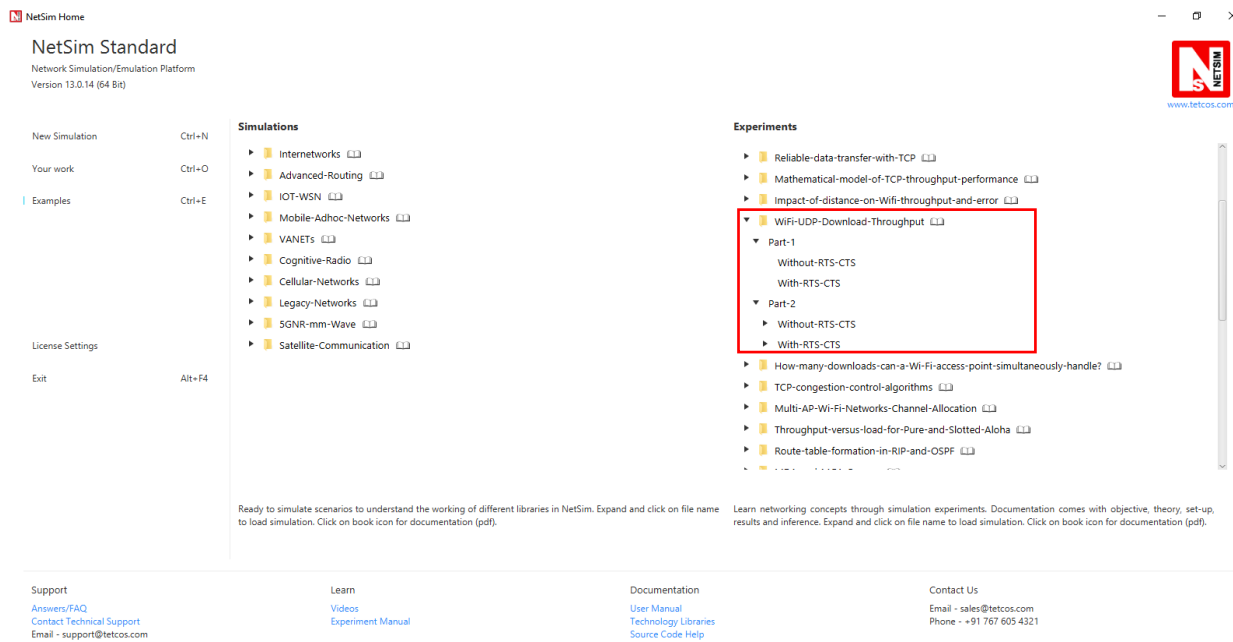


Figure 11-4: Experiments List

NetSim UI displays the configuration file corresponding to this experiment as shown below  
**Figure 11-5.**

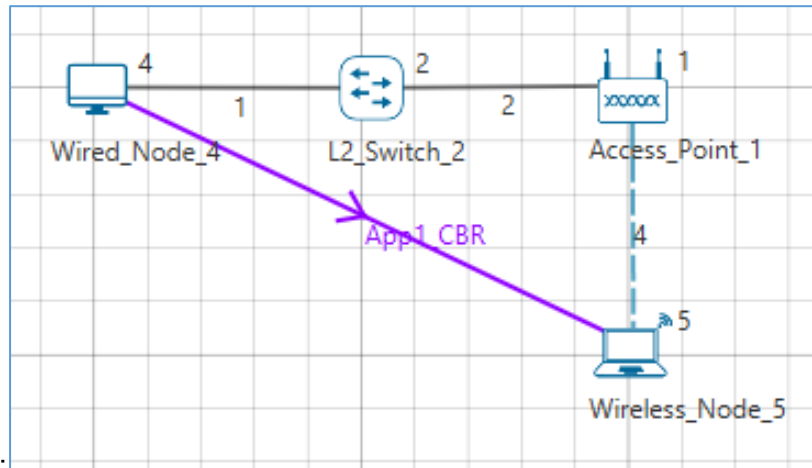


Figure 11-5: A single AP-STA Without RTS/CTS

## 11.4 Procedure

### 11.4.1 Part-1: Without RTS/CTS

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

**Step 1:** A network scenario is designed in NetSim GUI comprising of 1 Wired Node, 1 Wireless Node, 1 L2 Switch, and 1 Access Point in the “**Internetworks**” Network Library.

**Step 2:** In the Interface Wireless > Physical Layer Properties of Wireless Node 5, Protocol Standard is set to IEEE 802.11b. In the Interface Wireless > Data Link Layer Properties of Wireless Node and Access Point, RTS Threshold is set to 3000. Medium Access Protocol is set to DCF for all nodes.

**Step 3:** In the Wired Link Properties, Bit Error Rate and Propagation Delay is set to 0 for both the links.

**Step 4:** In the Wireless Link Properties, Channel Characteristics is set to NO PATH LOSS.

**Step 5:** Right click on the Application Flow **App1 CBR** and select Properties or click on the Application icon present in the top ribbon/toolbar. A CBR Application is generated from Wired Node 4 i.e., Source to Wireless Node 5 i.e., Destination with Packet Size set to 1450 Bytes and Inter Arrival Time set to 116 $\mu$ s. Transport Protocol is set to **UDP**.

The Packet Size and Inter Arrival Time parameters are set such that the Generation Rate equals 100 Mbps. Generation Rate can be calculated using the formula:

$$\text{Generation Rate (Mbps)} = \text{Packet Size (Bytes)} * 8 / \text{Interarrival time } (\mu\text{s})$$

**Step 6:** Plots are enabled in NetSim GUI. Run the Simulation for 10 Seconds and note down the throughput.

### 11.4.2 Part-1: With RTS/CTS

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

**Step 1:** In the Interface Wireless > Data Link Layer Properties of Wireless Node and Access Point, RTS Threshold is set to 1000.

**Step 2:** Plots are enabled in NetSim GUI. Run the Simulation for 10 Seconds and note down the throughput.

### 11.4.3 Part-2: Without RTS/CTS: 2APs

The following changes in settings are done from the previous sample **Figure 11-6**.

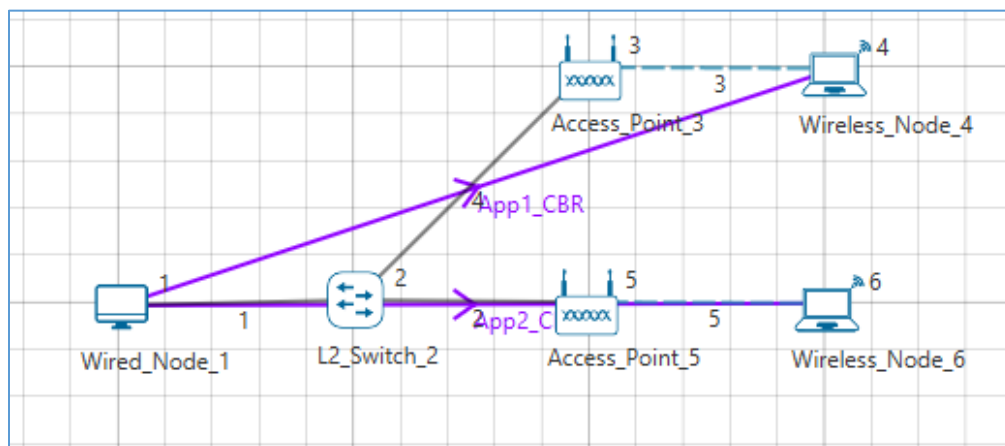


Figure 11-6: Multiple AP-STA Without RTS/CTS

**Step 1:** A network scenario is designed in NetSim GUI comprising of 1 Wired Node, 2 Wireless Node, 1 L2 Switch, and 2 Access Points in the “**Internetworks**” Network Library.

**Step 2:** In the Interface Wireless > Data Link Layer Properties of Wireless Node and Access Point, RTS Threshold is set to 3000. Medium Access Protocol is set to DCF for all nodes.

**Step 3:** Two CBR applications are generated from Wired Node 1 i.e., Source to Wireless Node 4 and Wireless Node 6 i.e., Destination with a Generation Rate of 10 Mbps.

**Step 4:** Plots are enabled in NetSim GUI. Run the Simulation for 10 Seconds and note down the throughput.



Similarly, the subsequent samples are carried out with 3, 4, and 5 Access Points and Wireless Nodes.

#### 11.4.4 Part-2: With RTS/CTS: 2APs

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

**Step 1:** In the Interface Wireless > Data Link Layer Properties of Wireless Node and Access Point, RTS Threshold is set to 1000. Medium Access Protocol is set to DCF for all nodes.

**Step 2:** Plots are enabled in NetSim GUI. Run the Simulation for 10 Seconds and note down the throughput.

Similarly, the subsequent samples are carried out with 3, 4, and 5 Access Points and Wireless Nodes.

Note: For the Next sample Newly added devices and links change the properties are per Without RTS/CTS example except RTS Threshold value.

### 11.5 Output I

After running simulation, check throughput in Application metrics as shown in the below screenshot **Figure 11-7**.

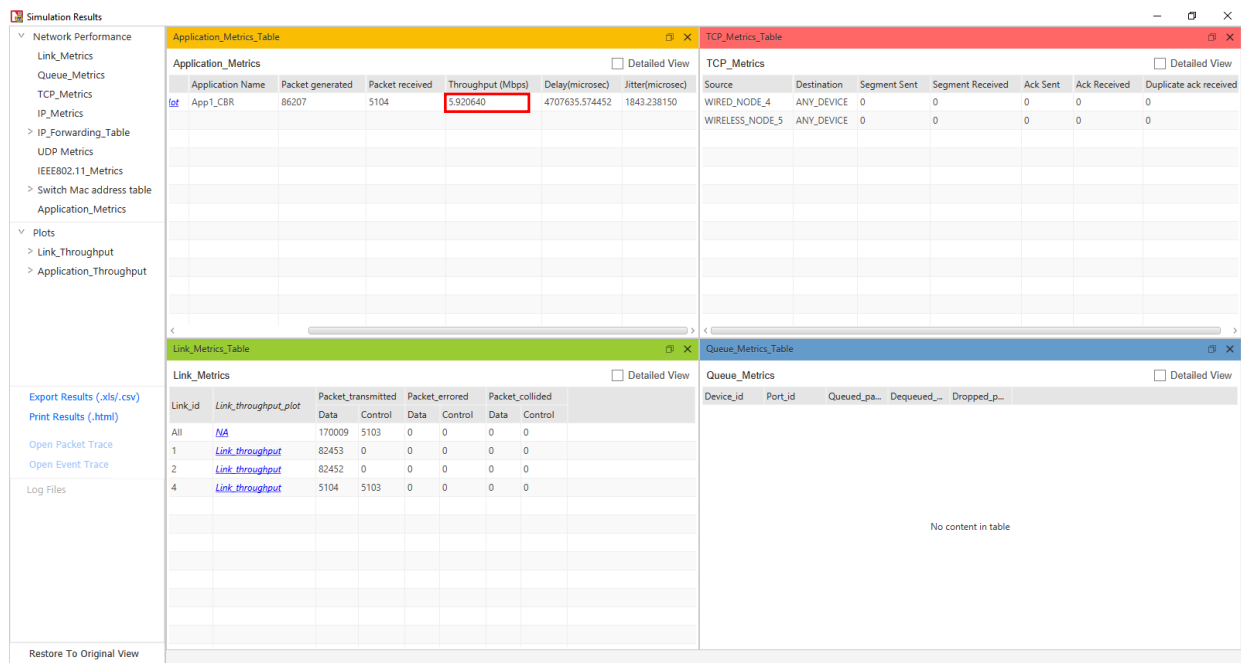


Figure 11-7: Results for Single AP-STA

Sample	Predicted Throughput (Mbps)	Simulated Throughput (Mbps)
<b>1 (Without RTS/CTS)</b>	5.89	5.92
<b>2 (With RTS/CTS)</b>	4.45	4.39

Table 11-1:UDP throughput for a single AP-STA, with and without RTS/CTS

## 11.6 Output II

After running simulation, check throughput in Application metrics as shown in the below screenshot **Figure 11-8**.

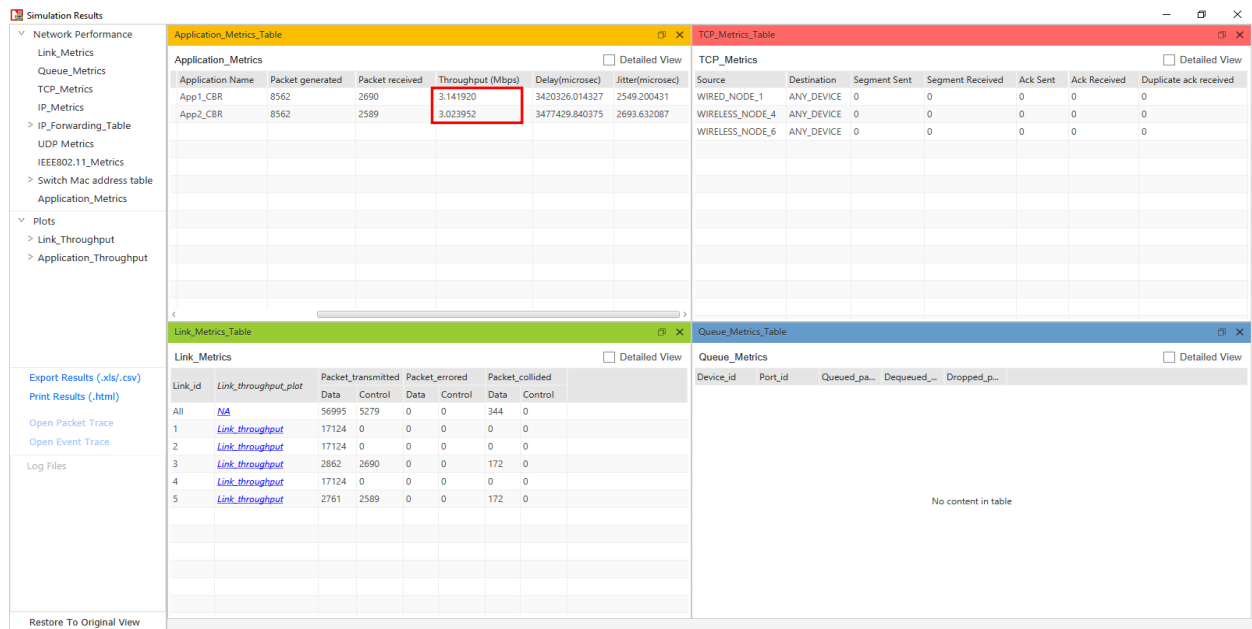


Figure 11-8: Results for Multiple AP-STA

Sample	Throughput (Mbps) with 2 APs	Throughput (Mbps) with 3 APs	Throughput (Mbps) with 4 APs	Throughput (Mbps) with 5 APs
<b>1 (Without RTS/CTS)</b>	App 1: 3.14 App 2: 3.02 <b>Total: 6.16</b>	App 1: 2.12 App 2: 1.97 App 3: 2.09 <b>Total: 6.18</b>	App 1:1.58 App 2:1.54 App 3:1.52 App 4:1.48 <b>Total: 6.12</b>	App 1: 1.25 App 2: 1.21 App 3: 1.19 App 4: 1.19 App 5: 1.24 <b>Total: 6.08</b>
<b>2 (With RTS/CTS)</b>	App 1: 2.34 App 2: 2.25 <b>Total: 4.59</b>	App 1: 1.58 App 2: 1.49 App 3: 1.58 <b>Total: 4.65</b>	App 1: 1.21 App 2: 1.18 App 3: 1.15 App 4: 1.12 <b>Total: 4.66</b>	App 1: 1.01 App 2: 0.90 App 3: 0.93 App 4: 0.92 App 5: 0.94 <b>Total: 4.70</b>

Table 11-2: UDP throughput for 2, 3, 4, and 5 AP-STA pairs, with and without RTS/CTS.

## 11.7 Discussion

**Table 11-1** shows the AP-STA UDP throughput (predicted and simulated) for a single AP-STA. **Table 11-2** shows the UDP throughputs for 2, 3, 4, and 5 AP-STA pairs; the total throughput is shown along with the individual AP-STA throughputs. We can make the following observations, along with explanations (as bulleted comments) for the observations.

1. The UDP throughput with RTS/CTS turned off is larger than when RTS/CTS is used.
  - The reduction in throughput with RTS/CTS is due the RTS/CTS overheads. The RTS/CTS mechanism aims at alerting “hidden” nodes that a transmission is about to start and can reduce collisions if there are hidden nodes. Since in this experiment all nodes can directly hear each other’s transmissions, the Basic Access mode suffices, whereas RTS/CTS only adds overhead.
2. The UDP throughput increases slightly with two AP-STA pairs than with just one.
  - With just one AP-STA pair, there is wastage of time due to backoffs, even when there is no possibility of contention. When one more AP-STA is added some of this wastage is compensated by two APs attempting, with the possibility that one of them might finish its backoff early and grab the channel, thus reducing the backoff overhead. There is, of course, the additional time wasted due to collisions, but the balance between these two opposing phenomena is such that there is a small gain in throughput.
3. Further increase in the number AP-STA pairs leads to a decrease in throughput, but the decrease is small.
  - The IEEE 802.11 Distributed Coordination Function (DCF) manages the sharing of the WiFi channel in a distributed manner. If there was a centralized scheduler than each AP could be scheduled by turn, without any backoff and collision overheads, and the total throughput would have been just that due to sending UDP packets back-to-back:  $\frac{1450 \times 8}{1296} = 8.95$  Mbps. Thus, the total throughput with DCF is smaller than if the UDP packets were being sent back-to-back, about 6 Mbps rather than 8.95 Mbps. However, DCF implements an adaptive attempt rate mechanism, which causes nodes to attempt less aggressively as the number of contending nodes increases. It is this mechanism that prevents the total throughput from dropping steeply as the number of AP-STA pairs increases.
4. The total throughput is distributed roughly equally between the AP-STA pairs.

- This is another feature of DCF. The contending nodes obtain fair access at the packet level, i.e., each successful packet is from any of the contending nodes with equal probability. The downside of this feature is that if an AP-STA is using long packets, then that UDP flow will get a larger throughput. In this experiment, all the AP-STA UDP flows are using the same packet lengths.