User guide for the Atheros CSI live view tools

This CSI live view tools is developed based on the open source project Atheros CSI Tool (<http://pdcc.ntu.edu.sg/wands/Atheros/>). We add the socket function to transmit the collected data to a remote server. At the server we use python to plot the CSI at real time.

This Tool has two separate parts: for our equipment:

‘Client’ need to be run on the PC which collect CSI and transmit the data to server through UDP;

‘Server’ need to work on another computer which run python to receive the CSI data and plot out. The client need to connect to the server through Ethernet (wire or wireless).

Hint: because of the requirement of the Atheros CSI tool, we need two computers which both use Atheros WiFi card and have installed with the customized kernel. One of clients needs to install Hostapd to work as an AP. Another client connects to the AP. That is to say, for the measurement we need two PCs, and for the plotting we need another computer.

After the hardware and software correctly installed, copy the whole folder ‘Client‘ to PCs (USB stick? Or Git repository).

One of the client needs to connect to the server through cable to transmit the CSI packet by UDP. Here we assume that 2nd computer(client) connect to server.

Configure one of the client which has installed hostapd as an AP, Run those commands:

1. ifconfig wlan0 down
2. ifconfig wlan0 10.10.0.1/24
3. service isc-dhcp-server restart
4. ./hostapd ./hostapd.conf & // don’t forget & to let the AP process run at background

If not, we could not run other command at this shell.

1. (After connection)
2. Go to ‘sendData’ folder

and run: ./send\_Data wlan0 4c:5e:0c:15:52:25 10000 //mac-address and number of packet may need to be changed

another client connects to the AP:

1. ifconfig wlan0 up
2. iw wlan0 connect Atheros\_csi\_tool
3. dhclient wlan0
4. goto ‘recvCSI-with-socket’ folder
5. and run: ./recv\_csi 192.168.199.120 5563 //the server’s ip and port may need to be changed

hint: the receiver(client) has three usages:

1. ./recv\_csi name.dat //save the collected CSI in local harddisk;
2. ./recv\_csi ip port //send the received csi to a server through udp
3. .recv\_csi ip port name.dat //send and save

hint: useful command: ifconfig to look at the ip address and mac address

If you use another type of computer (for example a laptop with GNOME), maybe you could connect to the AP through configuration on the GUI.

when you apply an ip through dhclient wlan0, maybe you receive this error:

RTNETLINK answers: File exists

sh: echo: I/O error

sh: echo: I/O error

sh: echo: I/O error

sh: echo: I/O error

This error means you already have an ip. So run this command to remove the old ip:

ip addr flush dev wlan0

At the server, there is an example: Atheros\_readandplot\_in\_realtime.py

1. If you want to read an offline data, which is logged at client or saved at server:

* Run this command: ret = Atheros.read\_from\_file('./withantenna.dat')

All the CSI\_matrix will be put in the dictionary: ret. You could do your things now as you wish.

1. Plot the CSI amplitude/phase at real time:
   1. s = udp.udp\_init(5563) //initialize udp, port could be manually changed
   2. data, addr = udp.recv(s) // receive udp packet
   3. stream = Atheros.read\_from\_stream(data) //decode udp packet, after this step, one CSI-matrix will save in stream
      1. Atheros\_plotandsave.amplitude(stream) //plot the amplitude
      2. Atheros\_plotandsave.phase(stream) //jplot the phase
   4. udp.close(s) //after use, udp must be closed
2. save the received CSI to file:
   1. s = udp.udp\_init(5563)
   2. filehandle = Atheros\_plotandsave.save\_to\_file\_init('udpsavetest.dat') //here input a file name in which you would store the CSI
      1. ret = [] //you could store the CSI only in register
   3. data, addr = udp.recv(s)
   4. stream = Atheros.read\_from\_stream(data)
   5. Atheros\_plotandsave.save\_to\_file(data, filehandle) //this step save one CSI row data in the file
      1. ret.append(stream) //store the CSI\_matrix in the dictionary
   6. Atheros\_plotandsave.close(filehandle)

The command of save-to-file and plot-CSI-at-realtime could work together.