allplot Command & Instruction Manual

This program is a python script that is used to analyze the ALL.TXT files generated by WSJT-X software. These files contain reception reports based upon the received SNR of the station heard. One of the main uses of this program is to compare the reception capabilities of two antennas when hearing the simultaneous signal for the same source. Hence, the usual method for using the program requires two ALL.TXT files recorded simultaneously, usually using a different antenna or receiver.

The program has a simple command structure. Here are the listed commands that you get if you issue a **H**ELP command.

```
Commands:
FA  Specify the A data file
FB  specify the B data file
CF  Combine Files - Find simultaneous matches
PO  POlar plot - process signal matches
BA  specify the Bands for data plot
AR  Animate Raw ALL.TXT file(s) on map
AB  animation of SNR A - SNR B data on map
PR  Polar plot against a Reference
MS  Map Signals - SNR A-B
MR  Map compared to A reference: B-A - C-A => B-C
CX  set color scale Maximum SNR A-B
CN  set color scale Minimum SNR A-B
SX  set polar plot scale Maximum
SN  set polar plot scale Minimum
ST  status
DP  set figure and animation resolution DPI
FS  set figure size, x-inches, y-inches
FT  turn on/off curve fit, F+/F-, Fourier degree, N (int)
DT  set time between frames for animation (minutes)
IT  set data collection time for frame (minutes)
LO  specify longitude, Latitude of observer
MX  set max SNR for interval to reject signals
VW  plot view: WOrld, NAmerica, AMericas, NHemisphere
PL  Plot Output: I+ inline console; S+ saved to file;  I-, S- turn off
XM  print out XML data
XD  reset allplot.xml to defaults and quit
Q   Quit
H   Help - this info...
```

-> is the **allplot** prompt

Running allplot in a python environment

Use the Anaconda Prompt for the allplot build environment, e.g. **Anaconda Prompt (allenv)** found in the Windows start menu under Anaconda3. Move to the allplot directory:

```
(allenv) C:\Users\You>cd allplot
(allenv) C:\Users\You\allplot>
```

To invoke the program:

```
(allenv) C:\Users\You\allplot>python allplot.py
```

Work Flow Summary

The two letter commands let you quickly process the files in the way you want. Typical work flow involves the commands in the following sequence:

->FA Get the first ALL.TXT file and enter a brief description.
->FB Get the second ALL.TXT file and enter a brief description.
->CF Combine the files by finding simultaneous reception matches.
->PO Generate a polar plot of the difference in SNR between the received signals
->BA 20m Select a desired band from the data.
->MS Map the SNR differences on the globe

If the data contains several bands, each band can be examined in turn with polar plots **PO**, and relative signal strengths, **MS**

Reception reports from a single file or a pair of the raw files can be animated as on a global map using **AR**.

Best for a single band, the difference in SNR between the two receivers can be animated on the global map using **AB**. Beware that this is noisy because no time averages are computed.

Two pairs of comparison files can make use of the first file, **FA**, as a reference. The **A** file for each pair must have the same description. For each pair of files, go through the steps to combine the files, **CF**, and process the matches by grid that happens with **PO**. At this point the **PR** command will generate a polar plot of the differences in grid-square averages between the **B** and **C** files (the *first pair's **B*** file becomes the **C** file). Similarly **MR** will map the SNR **B-C** differences on the globe. The data is noisier that with simultaneous measurements, but some trends can be obtained this way when information about a single antenna may be desired under different conditions by comparing to a fixed reference antenna.

Inline and Saved Plots, Status

Inline plots, polar plot or maps, will show up in a plot window when commanded. You must exit the plot window to continue with the program. If you wish to save the plots, turn them on(off) with **PL S+(-)**. Similarly you can turn off the inline plots, so you don’t have to close the windows all the time, with **PL I-.**

The program status can be checked with the **ST** command. Typical output:

```
ALLPLOT version 1.1   Built 2022-12-14  by Gary Rondeau, AF7NX

MapView: World View

Working files:
A: InvL   C:/Users/Gary/allplot/InvL0111.TXT
B: Loop   C:/Users/Gary/allplot/FlagN0111.TXT
Saved plots PATH:  C:/Users/Gary/allplot

Inline plots are ON   Saved plots are OFF
Files already matched and processed
160m  band selected

No valid reference antenna
```

Missing Grids

Sometimes a station you would really like to include in the data never transmits its grid location. If you can manually determine the grid location from another source, you can add the call and grid to the ***Grids.csv*** file. Just manually edit this file and include call and grid. If missing grids are encountered, they will be filled in from this file if possible. If you want to find out what you are missing, check the ***BadGrids.csv*** file for the list.

Paths and directories

Various temporary working files are created in the working directory where allplot.py resides when running the program. The working folder should contain the dependencies listed below.
Plots are saved to the directory where the **‘A’** data file resides. Setting up a directory for a particular experimental run is recommended.

HDSDR Schedule Band Changes

When doing studies involving more than one band at a time it is very nice to have some way to synchronously change the frequency band. One method is to use the schedule feature in the HDSDR program. HDSDR has a “CAT to HDSDR” mode where the program will send control commands to an attached radio to change frequency, etc. You do not need the program to do anything else; it need not receive any signals – it just needs to switch the frequency of the main receiver(s) at the commanded intervals. The distribution files on Github include a few HDSDR schedule files that can be used to switch bands. These files were generated with the ***HDSDR_Sked.py*** program, also in the distribution. For comparing two or more antennas, multiple instances of HDSDR can be run simultaneously on one computer, or could be run on at cooperating partner site on a different computer. If the same HDSDR schedule file is used, the band changes will all be synchronous so that the ***ALL.TXT*** files generated in WSJT-X will simultaneous received signals for them multiple bands defined by the schedule.

The HDSDR schedule feature is designed to be able to record material off the air and will generate an audio file for each band change. The schedule limits the record length to only 2 seconds, but still a WAV file is generated and these files can quickly fill up your disk drive – so beware. They need to be deleted on a regular basis.

Local Dependencies

allplot.py	the main script
config.py	the XML congiurator
allplot.xml	XML configuration file

Python Packages used

pandas
numpy
symfit
cartopy
BeautifulSoup
ffmpeg-python

See the installation instructions for details.

Bugs and Code Editing

I am sure there are a few bugs and plenty of missing features. Please report bugs via the contact link on my web page at ***www.squashpractice.com***
For simple editing and code inspection you can use a good editor such as Notepad++. For serious programming and debugging the Spyder IDE in the Anaconda distribution is the way to go.

Compiled allplot.exe (doesn’t exist yet)

allplot has **not** been successfully complied into as stand-alone distribution on Windows systems using the Nuitka compiler **yet**. The compiled version runs at least twice as fast on CPU-intensive automation generation. The down-side is that the distribution is very large, approximately 1.2 GB for the distribution folder.

When running the compiled version, run ***allplot.exe*** from the working directory, ‘allplot.dist’.

allplot Commands

AB Generates animated map of SNR differences. Requires a pair of matched (**CF**) and processed (**PO**) files. Best for a single band since color scale has no band information otherwise.

AR Generates animated map of raw signal report occurrences. Requires one or two input files.

BA *bnd* Selects band for data to plot. valid options: *bnd* = '10m', '12m', '15m', '17m', '20m', '30m', '40m', '80m', '160m'. Empty response, <CR>, or invalid entry selects 'all bands'.

CF Combine Files -- takes the A and B files and finds all of the simultaneous reception reports. Temporary file, Matches.csv is saved.

CN *mn* Sets minimum value for colorbar plots, *mn* in decimal dB. Empty response checks for minimum value of data set and sets that value.

CX *mx* Sets maximum value for colorbar plots, *mx* in decimal dB. Empty response checks for maximum value of data set and sets that value.

DP *dpi* Sets the "dots per inch" resolution for the graphics, *dpi* integer.

DT *t* Sets the time between frames for animations, *t* integer minutes. Default 3 minutes.

FA Select the 'A' ***ALL.TXT*** raw data file. You will be prompted for a short description as well.

FB Select the 'B' ***ALL.TXT*** raw data file. Again, you will be prompted for a description.

FS *w h* Sets the figure size in width *w* and height *h* in inches. Overall number of pixels is dpi^2 x *w* x *h.*

FT *[str][n]* Turns on/off, *str* = 'F+'/'F-', curve fitting for polar plot data. The curve fit is with a Fourier series with *n* = order.

H Help – prints out the command list.

IT *t* Sets the integration time *t*, in minutes, over which data is collected to be displayed in a single frame for animations.

LO *lon lat* Home longitude and latitude, determines bearing angles and central location of map plots.

MR Map SNR differences from a reference antenna for one antenna in two configurations. Requires a saved ***Lastmatchsort.csv*** file from previous antenna pair, and requires that antenna 'A' descriptors be identical. Use **CF**, **PO** on the pairs of files first. Then use **PR** to generate polar plot, and lastly **MR**.

MS Map SNR differences between two antennas. First use CF and PO to process the data, then MS.

MX Sets the maximum SNR report value during a transmission period to still accept the other data during that period. Default is 40, which never happens. Use this on radios that easily overload on strong signals.

PL *str* Sets how plots are displayed and saved. *str* = 'I+' turns on inline console plots; *str* = 'S+' turns on plots saved to a file; 'I-', 'S-' turn these off.

PO Processes the matched signal reports and generates a polar plot of the SNR differences.

PR Generates a polar plot for two configurations of an antenna compared to a reference antenna. Requires a saved ***Lastmatchsort.csv*** file from previous antenna pair, and requires that the antenna 'A' descriptors be identical. Use **CF**, **PO** on the pairs of files first, then **PR**.

Q Quit the program. Also ^C works.

SN *mn* Set the minimum scale marking for polar plots, *mn* in dB.

ST Prints out program status info, including program version and build date, to the console.

SX *ms* Set the maximum scale marking for polar plots, *mx* in dB.

VW *str* Sets the extent of the map plot view. *str* = 'WO' World; 'NA' North America; 'AM' North and South America; 'NH' North hemisphere.

XD Resets ***allplot.xml*** to start-up default and quits.

XM Lists the ***allplot.xml*** file to the console.