Russell Silva
8/5/2018 – 8/11/2018
AMRUPT, Summer '18

Report #5 – Bulk and Phase Offsets, Direction Finding on GNU Radio

Overview
- Confirmation of Bulk Offset Correction
- Minimization of Phase Drift
- Using the Noise Source for Phase Offset Correction
- Correction for Antenna Gains/Phases using a Pilot Beacon
- Preparation for AoA Testing

Confirmation of Bulk Offset Correction
Based on the conducted cross correlation tests, the sample offset blocks have corrected timing offsets. When the noise source is turned on and sample offset blocks are applied to the received signal, the timing offset goes to zero as shown in the GUI Vector Sink.

Minimization of Phase Drift
The phase difference between RTL SDRs drifted rapidly during initial tests. When dithering was turned off from the RTL SDR source block code, there was no phase drift between the RTL SDR data streams. The phase drift of a common signal source remained constant to less than a tenth of a radian over a ten-minute test.

It is important to note that the following commands must be entered at the terminal in order for the RTL SDR source blocks to function in GNU Radio:

export LD_LIBRARY_PATH=/usr/local/lib
gnuradio-companion

Using the Noise Source for Phase Offset Correction
Based on the conducted phase alignment tests, the cross correlation procedure with the noise source on has not corrected phase offsets from a common sinusoidal signal source. However, it is important to note that the noise source exhibits an unstable (+/- 0.2 radians) phase offset between channels after bulk offset correction. Furthermore, the offset between the noise source phase difference and the signal source phase difference was observed to be constant over multiple tests for multiple frequencies. When different combinations of RTL SDRs were tested, this offset between the noise source phase difference and the signal source phase difference changed. A possibility for coarsely correcting phase offsets is to observe the noise source and signal source phase offsets over multiple trials for multiple combinations of RTL SDRs. These observations would allow us to apply a constant when performing phase offset correction with the noise source. A possibility for the noise source's deviation from the actual phase difference is the extra channels (i.e. bias tees on the RF switches) the common signal must travel through each receiver

before reaching the noise source. Also, a phase difference was exhibited when only one RTL SDR was connected, which may be an indication of signal leakage between RTL SDRs.

Correction for Antenna Gains/Phases using a Pilot Beacon

A method in [1] to acquire sensor gain and phase uncertainties by eigenstructure analysis and decomposition has been adopted in the Ettus Blocks to correct for antenna phase offsets using a Pilot Beacon. It is important to note from [1] that a linear array would violate a necessary condition for uniqueness of a pair of sensor gains and phase values as a solution for the "set of sensor gains and phases and DOA angles." However, the documentation for the calibrate linear area GRC block states "This block calibrates a uniform linear antenna array. The input stream to this block is the sample correlation matrix generated using the signals received across the array elements."

To incorporate these blocks in real time, message passing [2] had to be implemented to enable/disable the calibration streams. Message passing allows for user to alter the state of the programs within the GRC blocks. This was implemented so that the user (and later the automated program) can choose when to phase calibrate the data streams (a constant pilot signal would interfere with the radio tag signals).

A known location beacon could also allow for phase offset correction by equalizing the phase difference between RTL SDR 1&2, 2&3, and 3&4. A carefully placed beacon could be put equidistant to two antenna array elements to zero out their phase difference. The phase differences between other elements in the array would then be equalized to the equidistant pair.

Flowgraphs for both beacon procedures, as well as a two-step procedure (with the noise source coarse phase offset procedure mentioned above), will be developed for AoA testing.

Preparation for AoA Testing

AoA tests will be performed on a field (i.e. baseball field with low multipath interference). If the previous test goes well, the system will be tested in a high multipath environment with uneven elevation (i.e. the botanical gardens). 0 and 90 degree angle reference points from the antenna array will be placed with pegs on the ground connected with 100-meter long strings. Then, 0-30-60-90-120-150-180-210-240-270-300-330 degree angles will be tested from the antenna array. All measurements will be performed using taut, lightweight strings and a protractor. Although simple, these tests may be more versatile than laser pointing in a highly forested environment with blocking vegetation and GPS testing (which will be inaccurate in itself).

Resources and relevant Forum Posts

[1] https://github.com/jakapoor/AMRUPT/blob/master/Literature/General_radio_direction_finding/Angle%20of%20Arrival%20Methods/Phase%20Offset%20Correction/A%20subspace%20method%20for%20estimating%20sensor%20gains%20and%20phases.pdf

[2] https://wiki.gnuradio.org/index.php/Message_Passing

GNU Radio Flowcharts Prepared for Testing

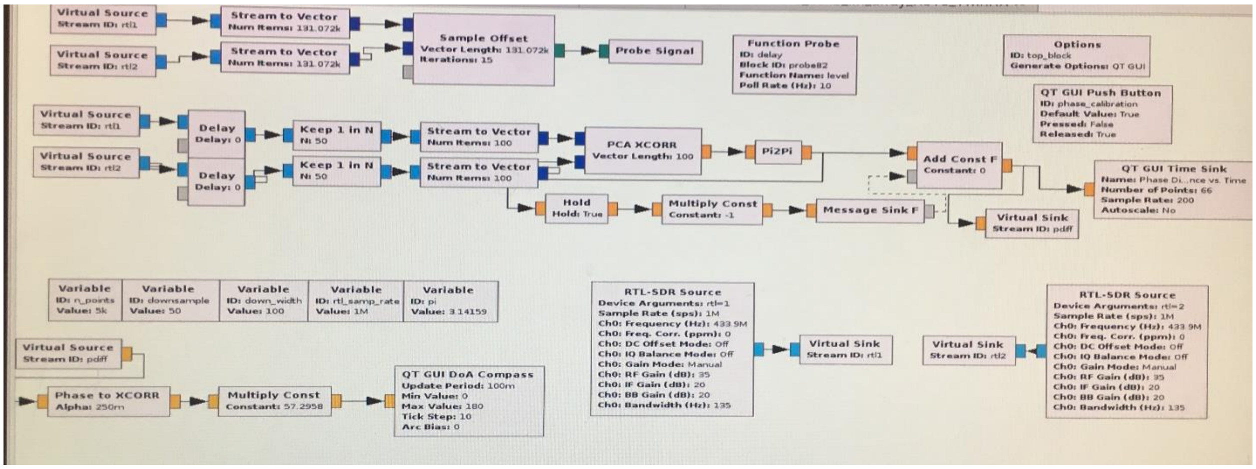

Figure 1: Angle of Arrival Flowchart using the Phase Difference Between Two Antennas

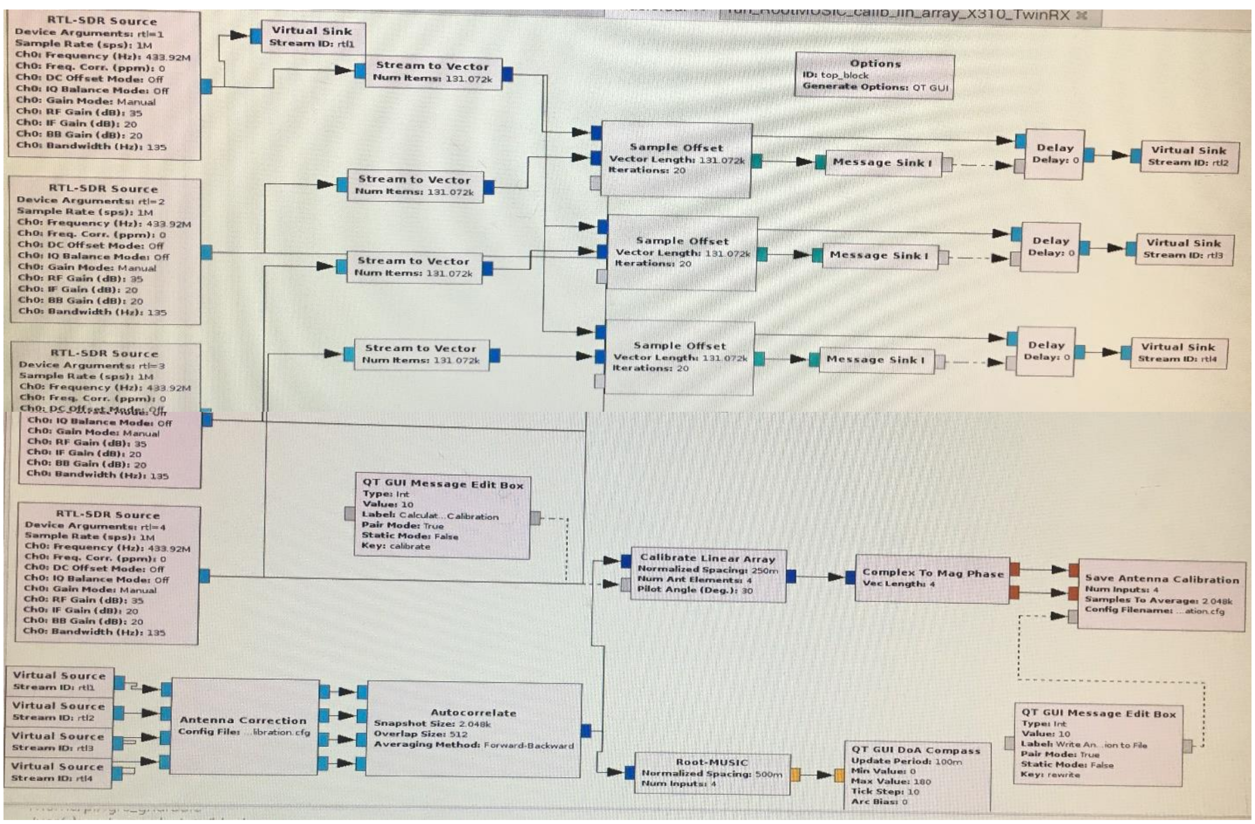

Figure 2: Angle of Arrival Flowchart using Root-Music and Phase Calibration with a Pilot Beacon. I still need to implement the sample offset delays as performed in Figure 1. I also must implement two autocorrelation streams that can be enabled/disabled (to prevent multiple phase

calibration and to mitigate runtime complexity. When running the above flowgraph, the Raspberry Pi can overheat, so 1 in N interpolation of the data-streams before autocorrelation and fractional autocorrelation (running autocorrelation on sub-windows of the incoming data) will be investigated as viable options.