AU Mic Notes

2016

Tasks

- Write up :/
- remake aumic_all
- package files nicely for Evan
- finish up modeling code, post to github

Someday / Maybe

- Make primary modeling code function-based

For Meredith

- "fits: insufficient information for antenna table"– is this a problem?
  - I'm pretty sure this only happened when I was getting a χ^2 of four... now that it's back to one, this error does not appear.
- χ^2 is back to 1– changing coordinates only affects χ^2 by 0.001 or less, so whatever the problem was it was not coordinates

---

13 January 2017

Cleaning up Modeling_Code, and deleted the following–noting it here in case it's needed later. 18aug2014 phasecenter='J2000 20h45m09.854710s -031d20m32.52034s' 24jun2015 phasecenter='J2000 20h45m09.867700s -31d20m32.89000s' 26mar2014 phasecenter='J2000 20h45m09.844300s -031d20m32.36000s'

---

11 January 2017

Quite accidentally, Meredith and I stumbled upon what was responsible for corrupting the visibilities. In order to ascertain how long the observations were on the flare date we plotted time vs. amplitude using uvplt (*uvplt vis=24jun2015_aumic1_spw3.corrected_weights.vis/ axis=time,amp device=/xs options=nobase*) and found that the last observation time window had become wonky, as seen in Figure 1. To fix this, I wrote a function to remove the last observation timewindow using the miriad command uvaver. The visibility file that uvaver spits out has one fewer index than the input file, so I added a conditional to my χ^2-finding function to accommodate this.

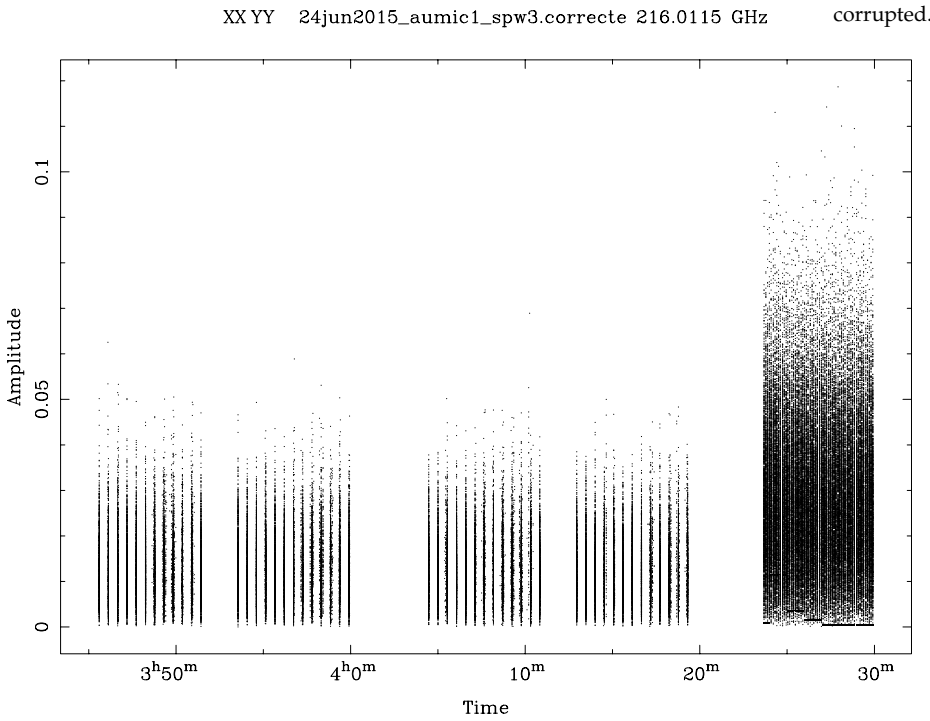

Figure 1: Amplitude as a function of time for the file with the worst χ^2. The final observation window is clearly corrupted.

18 November 2016

Weights

While looking at Kevin's weight correction code, Meredith and I realized that the code only calculates the weights for the *real* component of the visibilities, and does not calculate the imaginary weights. As such, I was applying the real weights to both the real and imaginary visiblities to obtain the χ^2 for my models. This is a decent approximation assuming that the real weights are roughly the same as the imaginary weights, i.e. **that the real dispersion is roughly**

the same as the imaginary dispersion. However, plotting the real weight vs. the imaginary weights implies that this is not the case, and regardless some accuracy is lost using this approximation. Instead, we are calculating the total weight for each point as

$$wt_{tot} = \sqrt{wt_{real} wt_{imaginary}} \tag{1}$$

and inserting the total weights into both the xx and yy polarization columns of each data file.

When calculating the corrected weights for each file using the method described above, the code prints the mean absolute difference between the real and imaginary weights, defined as

$$\mu_{diff} = \frac{\sum |wt_{real} - wt_{imaginary}|}{N}$$

The values for μ_{diff}, as well as χ^2s calculated with the corrected weights, are tabulated below.

File	Reduced χ^2	μ_{diff}
18aug2015_spw0	2.04	2777.94
18aug2015_spw1	2.04	3262.7
18aug2015_spw2	2.04	3354.83
18aug2015_spw3	2.05	3136.28
24jun2015_spw0	2.12	$1.041\,71 \times 10^6$
24jun2015_spw1	1.99	963318.0
24jun2015_spw2	2.01	$1.267\,27 \times 10^6$
24jun2015_spw3	12.18	$2.396\,07 \times 10^8$
26mar2014_spw0	2.07	4755.47
26mar2014_spw1	2.07	5064.5
26mar2014_spw2	2.06	5180.01
26mar2014_spw3	2.07	4431.92

Using Equation 1 reduced the reduced χ^2s for the bad spectral windows–this implies that the previously un-included imaginary weights tend to be smaller than the real weights. Kevin's weight correcting code also took about a factor of ten longer to run for the June date than for either of the other two.

---

1 November 2016

- Made sure that Modeling_Code_check.py deletes existing .vis files before it remakes them to avoid overwrite failure.
- Began setting up check code to do the splitting through miriad, rather than CASA– hopefully I can use the weights created by Kevin's code for the miriad-split visibilities.

25 October 2016

Created a new directory in AU_Mic titled "fixing_spws" to hold things related to fixing bad spws.

Created a specialized version of my modeling code, Modeling_Code_check.py, to check χ^2 for different spw splits.

- Splitting 24 Jun spws 2 and 3 (one good spw and one bad one) by time to compare χ^2
- NOTE: Can just use exportuvfits to split

References