December 6, 2006

Blanking and Flagging (a.k.a. ct005.py)

Lisa Wei, Peter Teuben (U Maryland)

See also: http://www.astro.umd.edu/~teuben/carma/blanking/

1.What this task entails: Historically blanking has been used by OVRO, while BIMA has only implemented flagging. Blanking is understood to mean that the data is thrown out of the time integration, whereas flagging merely flags the data with an associated boolean as being “good” or “bad”. In Miriad a “true” flag means the data is good. Carma will implement the best of both worlds: blanking is particularly efficient for long integration times, as small amounts of bad data can be blanked out without affecting the data quality. Flagging can be good for short integration times where the user can see data quality. The user will have considerable control over how blanking is turned into flagging (e.g. what triggers blanking, at what threshold blanks can become flags, etc.).

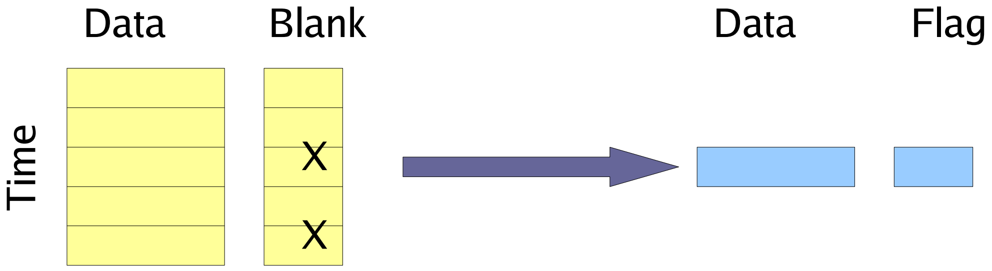

Fig 1: Transforming partially blanked (X) visibility data in the Carma pipeline to flagged data in Miriad.

2. What has been done: We first tested the response of (band 1) blanking on one OVRO and one BIMA antenna (CARMA 4,7) by setting trackTolerance = 0.0. The result was as expected: all the visibilities with baselines on antennas 4 and 7 were blanked in band 1 on antennas 1-8, the only place where blanking was allowed at the time (though also recall that because blanking is not implemented in Miriad, any blanking is currently implemented as a flagging).

We then experimented with turning blanking on and off in between scans in the script to ensure that blanking works properly when it is called upon. Carma 1, 4, 7, and 8 were used, this time. The observers at the time noticed that there was some dead time (~ 8 minutes) each time blanking is turned back on, which is reflected as holes in the data. Other than the dead time, the system blanked the data in band 1 on the proper scans. This problem appears to be resolved if the trackTolerance is set to a very small value, instead of exactly 0.0.

3. Quantitative results: none (see next section)

4. What remains to be done: At this point in the commissioning task, it was decided that until Carma code deals with the propagation of band-dependent integration times into Miriad datasets, the testing be put on hold. The most likely solution is to create a new Miriad header variable that reflects an integration time that depends on the band, BNDITIME(NSPECT). This will also need some change in the Miriad code.

Once this has been implemented in both Carma and Miriad, we envision the following kinds of tests:

1. Repeat the above described test, thresholding the pointing of an antennae a bit as to expect partial or full blanking. For given wind conditions make a histogram of the effective integration times. This also gives us an idea how well different antennae can track under given wind-conditions (speed/direction), which is probably also interesting for engineers. A full scale test of this effect needs knowledgeable observers.
2. Test band dependent blanking.
3. Test variance in timestamps

We will need a small (probably new) task in Miriad to display the integration times and timestamps per band/baseline. Mainly based on the expected BNDTIME(NSPECT) variable.

Notes:

1. Miriad does not have a weight associated with a visibility (unlike AIPS), only a flag. Weighting can be using options=systemp in INVERT. See also the options=varwt in the FITS program.
2. Need some example(s) to show how users could control the transformation from Carma to Miriad data in blanking/flagging.