ALMA Pipeline Heuristics

Atacama Large Millimeter Array

Dirk Muders (MPIfR), Frédéric Boone (LERMA), Friedrich Wyrowski (MPIfR), John Lightfoot (UKATC), George Kosugi (NAOJ), Christine Wilson (McMaster U), Lindsey Davis (NRAO), Debra Shepherd (NRAO)

ADASS XVI
P2.07
Tucson, AZ

Summary

The ALMA (Atacama Large Millimeter Array) Pipeline Heuristics system is being developed to automatically reduce data taken with the standard observing modes. The goal is to make ALMA user-friendly to astronomers who are not experts in radio interferometry. The Pipeline Heuristics must capture the expert knowledge required to provide data products that can be used without further processing.

Observing modes to be processed by the system include single field interferometry, mosaics, and single dish 'on-the-fly' maps, and combinations of these modes. The data will be produced by the main ALMA array, the ALMA Compact Array (ACA), and single dish antennas.

The Pipeline Heuristics system is being developed as a set of Python scripts using as the data processing engines the CASA/AIPS++ libraries and the ATNF Spectral Analysis Package (ASAP).

The interferometry Heuristics scripts currently provide a five stage process comprising flagging, initial calibration, re-flagging, re-calibration, and imaging of the target data. A Java browser provides user-friendly access to the Heuristics results.

Several techniques are used to search for bad data. In the spectral domain edge detection algorithms are applied, while in the time domain running mean methods are used. We have begun to develop methods to detect phase and gain jumps. Amplitude and phase gain statistics are used for re-flagging. The flagging algorithms have been re-organized to perform most of the tasks antenna- rather than baseline-based. Basic imaging parameters are determined automatically.

The initial single-dish Heuristics scripts implement spectral baseline fitting routines, which utilize a linefinder class in ASAP as a line detection engine of simple thresholding technique. By analyzing spatial extent and continuity along with the persistence of each detected feature, realistic scientific features are extracted and protected against baseline subtraction. Baseline characteristics are determined by a major component analysis in Fourier space.

The resulting data cubes are analyzed to detect source emission spectrally and spatially in order calculate signal-to-noise ratios for comparison against the science goals specified by the observer.

This poster describes the reduction datapath and the algorithms used at each stage, recent test results, and the path for future development.

The Mission

The goal of the Pipeline Heuristics team is to develop algorithms that will reduce ALMA data to give results of publishable quality automatically. Such performance is needed for 2 reasons:

- The large size of the raw datasets produced by ALMA will make them difficult to reduce by hand in the traditional way.
- We must make the instrument readily 'useable' to astronomers who are not (submm) interferometry or single-dish observation experts.

If we succeed then even experts should feel no need to re-examine the data in an effort to eke out a better result.

The Development Process

The dictionary definition of a 'heuristic' process is one where the solution is found through trial and error. As such the heuristic development of algorithms to reduce radio interferometry data has been going on since the first such instrument was built. Our main task is to capture the expertise that has built up in the community and form the reduction algorithms accordingly.

Our development model is incremental; we build a script to reduce simple data, test it, modify the script to handle more complex data, test it, and so on…. The idea is that at each stage we have a concrete product and that more 'expertise' can be incorporated with each development cycle.

The Expertise

- At a direct level expertise is provided by the scientists within the Pipeline Heuristics team.
- A second level has been obtained by some members of the team canvassing experienced interferometer users for their methods.
- A third level is provided by feedback from the independent testers who examine the results of each test in the development cycle.

The Tools

Comparative tests between several available packages showed that the AIPS++ tools provide the performance needed and were the most easily extensible to new capabilities.

The AIPS++ Distributed Object framework has been modernised by replacing the former Glish/C++ binding by a C++/CORBA binding that makes the tools available to Python, as CORBA objects within the ALMA Common Software (ACS). In the process the underlying AIPS++ libraries have been reorganised and renamed as Common Astronomy Software Applications (CASA/CASAPY).

ASAP (the ATNF Spectral Analysis Package) is being used for the reduction of single dish data.

The Next Steps

- Improve imaging heuristics.
- Explore ways of detecting and correcting data taken at times of rapid sky phase variation.
- Handle more complex line shapes for single-dish baseline fitting.
- Combine interferometry and single-dish data.

Interferometry Heuristics

The Current Interferometry Algorithm

The aim for the current development cycle is to produce an end-to-end reduction of the single-field observing mode. We have implemented a five stage algorithm to accomplish this:

1. Detect and flag bad data for the BANDPASS and GAIN calibrators.
2. Derive and apply the BANDPASS and GAIN calibrations.
3. Re-flag the calibrated data and derive the calibrations again.
4. Flag outliers in the target data.
5. Apply the calibrations and image the target data.

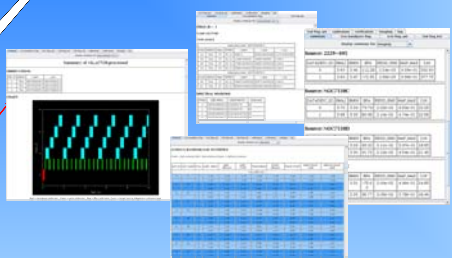
Data Assembly

For our current tests we are using CASA/AIPS++ MeasurementSets with data from the VLA and Plateau de Bure. Metadata for these datasets is provided implicitly.

In the delivery system the data will arrive conforming to the ALMA Science Data Model (ASDM) where Metadata will be held explicitly; an associated ParameterSet may specify optional reduction details e.g. angular resolution, velocity resolution, etc.

Check Dataset

The aim of this stage is to check that the dataset contains the necessary calibrators.



Flag Pass 1 (Raw Data)

- Flag autocorrelations.
- Flag data corrupted by known hardware quirks, e.g. PdB Gibbs channels.
- Flag data from shadowed antennas.
- Flag ends of bandpass where the bandpass profile is strongly curved.
- Apply a sliding median along the time axis of the data and flag statistical outliers.

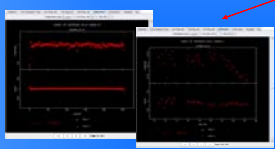
VLA: End channel flagging of bandpass calibrator and bandpass calibration

Calibrate the GAIN Calibrator

For BANDPASS calibrator:
Apply pre-measured instrument gains (if any)
Solve for GAIN
Solve for BANDPASS

For GAIN calibrator
Apply pre-measured instrument gains
apply BANDPASS calibration
Solve for GAIN

PdB: Calibration solutions



Flag Pass 2: Calibrated Data

- Flag antennas with anomalous gain
- Flag GAIN measurements that are statistically anomalous in the time sequence
- Apply a sliding median to the calibrated data along the time axis and flag outliers.

PdB: Flag calibrated data

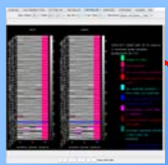
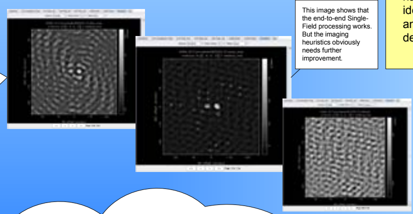


Image the target source

- Set the image centre, size and pixel size
- Select all data in the spectral window
- Set 'natural' weighting
- Make a clean image using the Clark algorithm

PdB: W3(OH) Dirty image, clean image and residual



This image shows that the end-to-end Single-Field processing works. But the imaging heuristics obviously needs further improvement.

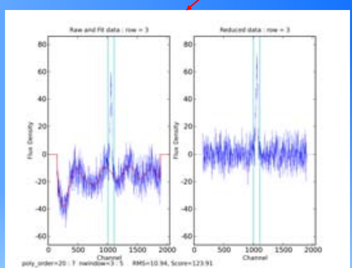
Single-Dish Heuristics

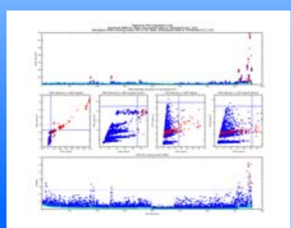
The Current Single-Dish Algorithm

The aim for the current development cycle is to automatically determine the baselines to be subtracted from calibrated spectral line data and to produce an image of the scientific target. We employ the following steps:

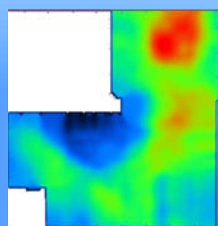
1. Fit and subtract the line emission to identify the line free sections of the spectrum to be used for the baseline reduction.
2. Fourier transform the residual and derive the baseline order from a major component analysis.
3. Remove baselines and grid data for the final image.

Determine line free sections and fit polynomial to residual baseline

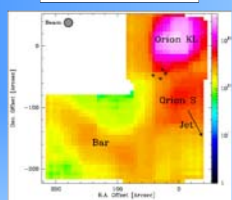




Remove baseline, calculate statistics and grid data



Comparison to published map (below) obtained from same data set shows good agreement



(T.L.Wilson et al., CO 7-6 in Orion KL)