The ALMA Pipeline Heuristics team is a group of astronomers who are not experts in radio interferometry. The Pipeline Heuristics must capture the specific steps and tools that are used in the interferometry data reduction process. The goal of this team is to make ALMA user-friendly to astronomers who are not experts in radio interferometry.

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 interferometry way.
- We must make the instrument readily accessible to astronomers who are not (radio) 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 dictionary definition of a ‘heuristic’ process is ‘one which utilizes a general approach for finding a solution but is not guaranteed to reach the best solution’. In this context, such the heuristic development of algorithms to reduce radio interferometry data has been ongoing 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 begin with a script to reduce simple data, test it, and so on... The idea is that at each stage we have a concrete product that can be incorporated with each development cycle.

The tools being developed of course will be tied to the interferometry process. While the interferometry needs a package to handle more complex data, this uses Python, as CORBA tools to handle more complex data.

The Next Steps
- Improve imaging heuristics.
- Explore ways of detecting and correcting data problems (e.g. baseline alignment, phase variation, parameter estimation).
- Develop line finder heuristics for single-dish baseline fitting.
- Develop interferometry and single-dish APIs.

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.

The Current Interferometry Algorithms

1. Detect and flag bad data for the BANDPASS and GAIN calibrators.
2. Derive and apply the BANDPASS calibration.
3. Remove baselines and grid data for the final image.
4. Flag outliers in the target data.
5. Apply the calibrations and image the target data.

Calibrate the GAIN Calibrator

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

Flag Pass 1 (Raw Data)

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

Flag Pass 2. Calibrated Data

- Flag features with arbitrary gain.
- Flag features with 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
PdB: W3(OH) Dirty image, clean image and residual
PdB: Calibration solutions

The Current Single-Dish Algorithm

The aim of the current development cycle is to produce a process to reduce the single-dish interferometry data. We have implemented a five stage algorithm to accomplish this:

1. Determine line free sections and fit polynomial to residual baseline.
2. Determine line free sections and fit polynomial to residual baseline.
3. Remove baselines and grid data for the final image.
4. Determine line free sections and fit polynomial to residual baseline.
5. Remove baselines, calculate statistics and grid data.

Several techniques are used to search for bad data in the spectral domain. The time 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 resulting data cubes are analyzed to detect source emission spectrally and spatially; in order calculate signal-tonoise 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.