rPICARD - A CASA-based Calibration and Imaging Pipeline for VLBI Data

14th EVN Symposium & Users Meeting
Granada, October 11 2018

Michael Janssen
BlackHoleCam* PhD student
Supervisors: Heino Falcke & Ciriaco Goddi,
Radboud University Nijmegen

*BHC is an ERC-funded project and partner of the Event Horizon Telescope Consortium
rPICARD - Radboud Pipeline for the Calibration of high Angular Resolution Data

- **AIPS**
  - Official support discontinued.
  - Steep learning curve for new python generation of astronomers.
  - Limited support for batch processing.

- **CASA**
  - Secure development future, ipython interface, built for batch processing → pipeline.
  - Widely used and well established (ALMA, VLA, …).
  - MPI scaling to deal with increasing data volumes in the future.
  - VLBI upgrade from JIVE+BlacHoleCam (Ilse’s talk)

- **rPICARD: CASA-based VLBI calibration and imaging pipeline**
  - Highly configurable & self-tuning parameters (e.g., fringe-fit solution interval based on SNR).
  - Verbose diagnostics (→ plots), easy to control and re-run + intervene semi-interactively.
  - Used for EHT data processing. Work with Ilse van Bemmel, Kazi Rygl, Elisabetta Liuzzo.
  - And for other arrays: GMVA, VLBA, EVN, … can work with any fits-idi files or measurement sets.
  - MPI speedup (fringe-fit scans in parallel).
  - Documentation/cookbook (40 pages).
  - Open source: https://bitbucket.org/M_Janssen/picard.
  - Science reproducibility.
rPICARD calibration

All plots shown are automatically generated by the pipeline
ANTAB Amplitude calibration and opacity fit

- \( T_{\text{sys}} \sim T_{\text{rx}} + (1 - e^{-T})T_{\text{atm}} \)
- \( T_{\text{sys}}^* = T_{\text{sys}} * e^T \)
- Find \( T_{\text{atm}} \) with Pardo et al. (2001) atmospheric code.
- Find \( T_{\text{rx}} \) by extrapolating \( T_{\text{sys}} \) to zero airmass.

7mm VLBA data of M87.
Project code: BW0106.
Fringe-fit solution intervals tuned by SNR

- CASA *fringefit* is copy from AIPS FRING task: Schwab and Cotton (1983).
  → FFT with SNR cut for initial guesses and station-based least-squares refinement.

- Skip least-squares for quick solution interval parameter search (smallest solint that yields detections on all possible baselines) per scan.

- Can have different solution intervals per station.

7mm VLBA data of M87 from June 2013.
Project code: BW0106.
Calibration Solution Examples (7mm VLBA)

7mm VLBA data of M87 from June 2013. Project code: BW0106. Two spectral windows (IFs).
Calibration Solutions applied (7mm VLBA)

Uncalibrated.

Calibrated. Edge channels flagged (crosses).

Plots made with Harro Verkouter’s *jplotter*.
rPICARD Calibration flowchart

- **Solid border:**
  - All sources used.

- **Dashed border:**
  - Calibrators used.

- **Dotted border:**
  - Science targets used.

- **Rectangular boxes:**
  - Applied to all sources.

- **Diamonds:**
  - Applied to calibrators.

- **Circles:**
  - Applied to science targets.
rPICARD imaging

Uses CASA tclean
Load the data. Use Importuvfits to import UVFITS files.

Use split to select the source and remove fully flagged data.

Do if input is set
- Apply flagtable.
- (stratmod_sc) with gaincal.
- (timeavg) with mstransform.

Set parameters
- Deconvolver.
- (Cellsize) & (imsize).
- (rms threshold).
- Data integration time \( t_{\text{int}} \).

Allow user to set (interactive) to False if (sautomated) is set to True.

T_{\text{amp}} = t_{\text{mp}} or iteration reached \( N_{\text{sciter}} \)?

Yes

T_{\text{clean}}.

delmod.

Plot latest diagnostics
- Image with imview.
- Self-cal solutions with plotcal.
- Data & model with plottms.

applycal: Correct the data.

Update \( T_{\text{amp}} \)
\( T_{\text{amp}} \) = (solint_denominator).

Update niter
niter += niter if (cleaniterations) is ‘shallow’.

Gaincal: phase self-cal on (phase_only_selfcal)[iteration] timescale.

Gaincal: amplitude self-cal on \( T_{\text{amp}} \) timescale for (amp_selfcal_ants) timescale.

Gaincal: phase self-cal on \( t_{\text{mp}} \) timescale.

No

Initialize run values
- niter = (niter0)
- Amplitude calibration timescale \( T_{\text{amp}} = \) startsolint.
- mask = (input mask).

Update niter
Niter += niter if (cleaniterations) is ‘shallow’.

T_{\text{clean}}.

(phase_only_selfcal)
iteration?

Yes

No

Exportfits.

Promising work from Jose Luis: Using auto-masking for automated imaging without user bias.

Loops of multi-scale tclean and self-calibration.
- Phases: accumulation period.
- Amplitudes: start at hours timescale and lower by factor of 2 in each iteration.

Stop after set number of maximum iterations or when calibrating amplitudes on accumulation period timescales.
Calibrated and imaged with CASA-based rPICARD pipeline.

Results agree with Walker et al. (2018):
- Weak counterjet.
- Edge-brightening.
- Large initial opening angle.
- Re-collimation of upper arm.

7mm VLBA data of M87 from June 2013.
Project code: BW0106.
Summary

- CASA is ready for VLBI.

- rPICARD, a first general purpose CASA-based calibration+imaging pipeline is available (Janssen et al., in prep.)
  - Promotes reproducibility of scientific results. Pipeline is open source: $ git pull https://bitbucket.org/M_Janssen/picard
  - Verbose diagnostics, self-tuning default parameters, highly configurable, interactive mode, option to quickly re-run steps.
  - MPI scalable.
  - Well documented.
  - Used for EHT data processing (cross-validated with AIPS and HOPS) and successfully applied to GMVA, VLBA, EVN, and synthetic data as well. Modularity → easy to add other arrays.
  - Phase referencing and polarization calibration supported (for leakage calibration a sufficiently compact calibrator is needed).
  - Future features
    - Spectral line calibration features (delay solutions from continuum sourced and rate solutions from the bright line).
    - LPCAL-like task to solve for leakage from calibrators with extended polarization structure.

- Coming soon: MeqSilhouette (Blecher et al., 2017) + rPICARD realistic synthetic data generation pipeline (Heino’s talk)
Run setup script to link CASA installation to rPICARD.
The setup script can also prepare a default set of input files for different arrays.

The next step is to copy your input files to the working directory.

Found
/home/michael/Software/CASA_builds_from_JIVE/casa-feature-CAS-10684-24.el7/bin/casa
as your CASA executable.
Checking this CASA version:

Has mpi: True
Has fringefit.py: True
Has accor.py: True

Press Enter and I will use the absolute path to this executable for picard.sh.
Write anything else (and then press Enter) to abort.
>

Writing the CASA executable path to a <your_casapath.txt> file, which will be used by picard.sh.

Making picard.sh executable.

Editing the input/mpi_host_file using the determined name of this computer (mjpc)
and 4 cores. Change this setup manually if desired.

I could put some default values for array.inp depending on which array you inted to use.

  0 for VLBAlo (for low frequencies)
  1 for EHT
  2 for VLBAhi (for high frequencies)
  3 for EVN
  4 for GMVA

Press enter without entering anything else to continue without altering your array.inp file. Else, enter the number corresponding to the array you want to use and press Enter

>3

The pipeline should be ready to run now.
If there are issues with mpiCASA contact M.Janssen@astro.ru.nl
or look at https://casa.nrao.edu/casadocs/@search?SearchableText=mpi

If you want to be able to run the pipeline from everywhere,
then you should add the following line to your .bashrc folder:
export PATH=$PATH:/home/michael/JeanLuc/Picard/picard/

Remember set some input parameters in the beginning, before running the pipeline.
At least edit input/observation.inp and input/array.inp

Please read documentation/picard documentation.pdf and follow the Quick Start Guide chapter to get started.
Typically, only science targets and calibrators have to be specified.
The start-up time of CASA may vary depending on whether the shared libraries are cached or not.

-------------------------------

The start-up time of CASA may vary depending on whether the shared libraries are cached or not.

-------------------------------

The start-up time of CASA may vary depending on whether the shared libraries are cached or not.

-------------------------------

The start-up time of CASA may vary depending on whether the shared libraries are cached or not.

-------------------------------

The start-up time of CASA may vary depending on whether the shared libraries are cached or not.

-------------------------------

IPython 5.1.0 -- An enhanced Interactive Python.
IPython 5.1.0 -- An enhanced Interactive Python.
IPython 5.1.0 -- An enhanced Interactive Python.
IPython 5.1.0 -- An enhanced Interactive Python.
CASA 5.3.0-136 -- Common Astronomy Software Applications
CASA 5.3.0-136 -- Common Astronomy Software Applications
CASA 5.3.0-136 -- Common Astronomy Software Applications
CASA 5.3.0-136 -- Common Astronomy Software Applications
CASA 5.3.0-136 -- Common Astronomy Software Applications

[Links to] files used in working directory (loaded automatically):
- Input folder
- Metadata
- Visibility data

MPI: start CASA N times
rPICARD will print the steps executed for the array to be calibrated (labeled for quick re-runs).

And the steps are executed.

The pipeline will execute the following steps for the EVN array in the given order:

- a: load models of observed sources (if present)
- b: use online flags from idi files (if present)
- c: use flags from metadata (if present)
- d: flag based on outlier detection from auto-correlations vs time
- e: flag based on outlier detection from auto-correlations vs frequency
- f: flag edge channels
- g: clear the calibrated data column of the MS from previous applycal runs
- h: apply all existing tables from all_calibration_steps
- i: print overview of flagged data (can be slow)
- j: make diagnostic plots of calibrated visibilities for selected baselines
- k: average and export the calibrated data

Can use quickmode [-q] to execute only a subset of these steps.

--- Executing step a ---

Loading model data for the observed sources...
Using
  /home/michael/JeanLuc/testrun/input/..3C84.smodel
as model for 3C84

Done

--- Executing step b ---

Getting flags from fits-idi files...
No FG table extension found in
  /home/michael/JeanLuc/testrun/input/..linkto_example_EVN.IDI1
Continuing without generating a flag file.

Done

--- Executing step c ---

Looking for metadata files with flagging information...
Found the following files with correlator/online flags:
  [/home/michael/JeanLuc/testrun/input/..example.flag]
Applying flags from metadata files.
Done applying metadata flags.

Done

--- Executing step d ---

Skip flagging based on autocorrelations vs time because flag_autocorr_vs_time is not set.