Joint Deconvolution (SDINT) Code

(Rau et al. 2019)

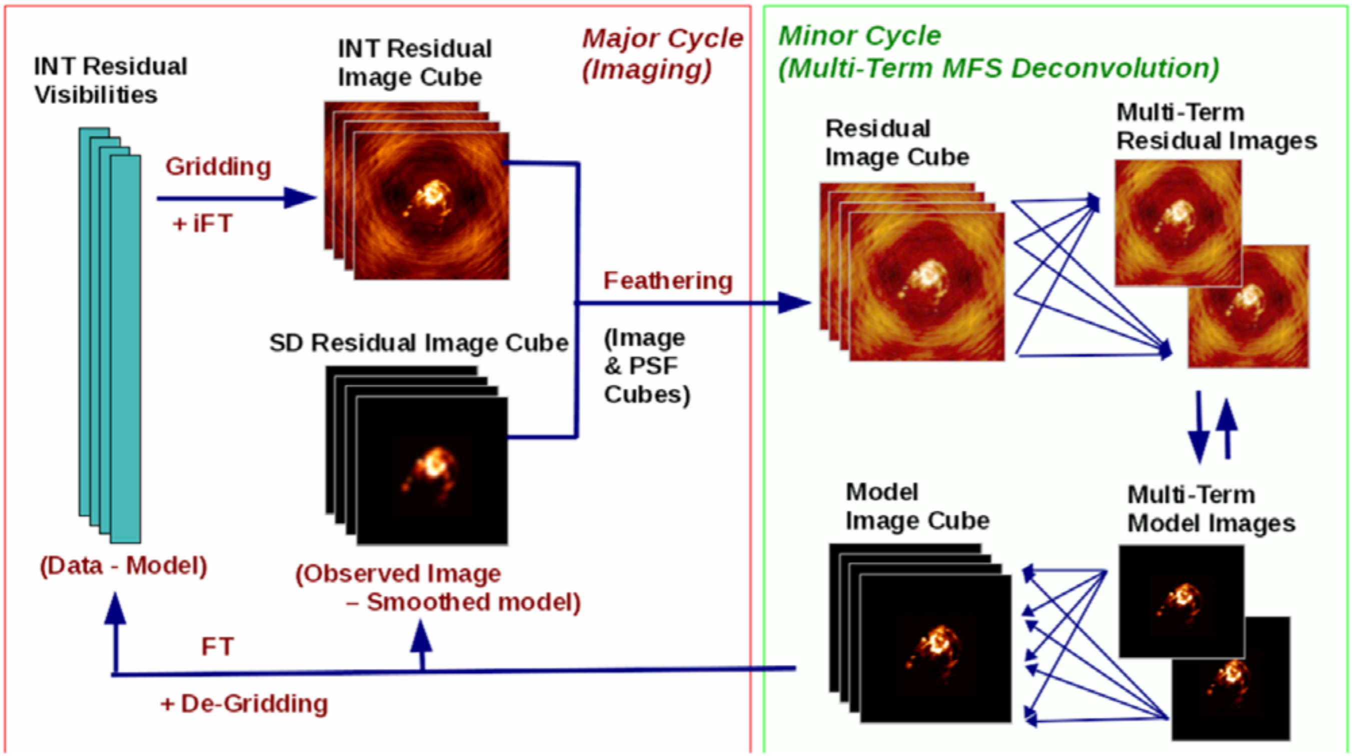
Improving Image Fidelity Workshop 2019
Lorentz Center, Leiden, Netherlands
Tim Braun

The JD Code

- Joint Deconvolution (SDINT) technique described in Rau et al. 2019.
- Feathers SD and INT data and psf's per channel prior to clean minor cycle to avoid burning in large-scale INT-only errors during clean.
- Works on both mosaic and single pointing INT wideband (or narrowband) data to mfs or cube image.
- Supports advanced imaging algorithms for cube or mfs imaging similar to tclean.

(See https://casa.nrao.edu/docs/TaskRef/tclean-task.html for a list of supported gridders, deconvolvers, & params.)

JD Algorithm Flow



The JD Code

JD code currently in three parts:

- *runsdint.py* :

Initializes parameters for tclean and feather and it defines input/output filenames.

- *sdint_imager.py* :

Uses PySynthesisImager to set up SD and INT imagers. Contains main loop for the joint deconvolution/joint image reconstruction where the iterative cleaning and feathering takes place.

- *sdint_helper.py* :

Contains all the pertinent helper functions for JD. Includes functions for feathering SD and INT cubes & psf's, primary beam math, creating residual cubes, calculating Taylor sum from cubes, etc…

Getting Started

- Get scripts from ScriptsforRealData on Urvashi's GitHub (linked to it on DC2019 GitHub page under Combination Techniques -> sdint)

- Ensure you have the following prior to starting scripts: INT MS, SD data cube (in Jy bm^{-1}), and SD psf cube (slide on this) with beam info in the SD data/psf headers.

- Fill in tclean, feather, and other params. you want to use in *runsdint.py*.

- If your favorite tclean param. is missing, then add it in *runsdint.py* and in *sdint_imager.py* as a class variable.

- Run code in casa window with: *execfile('runsdint.py')*

Aside: Creating SD PSF Cube

- Have to assume a beam shape and size. For simplicity, we can assume Gaussian for GBT.
- We can use CASA Toolkits (e.g., componentlist, image, coordsys) to create a Gaussian placed in the center of each plane in a data cube with a peak of 1 Jy.
- Simple python/CASA script available if needed.

Aside: SD & INT Spectral Axes

- If running in SDINT mode: **ensure the SD & INT channels are aligned and are of the same size**. (Run normal tclean with niter=0 on INT data. Rebin SD spectra to INT spectra and then regrid to align channels.
- This might blank some edge channels.

Comparing INT,SD,Feather,SDINT

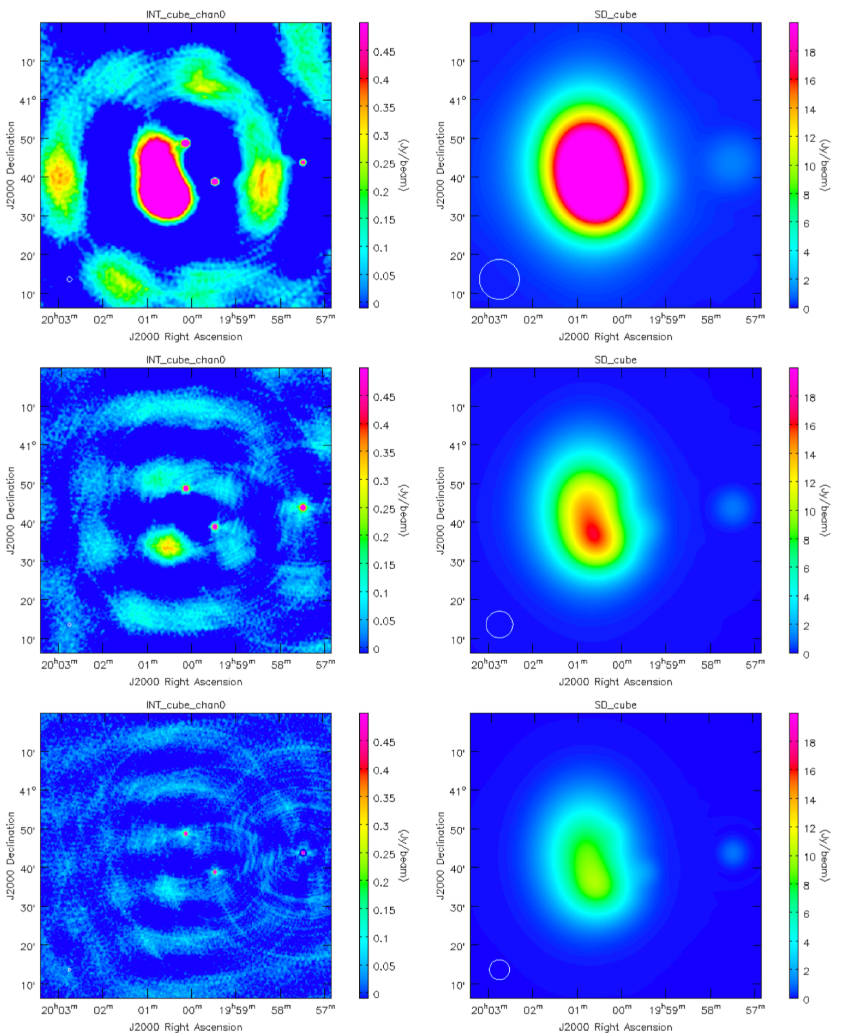


Figure 4. Deconvolved INT-only (LEFT) and SD-only (RIGHT) Spectral Cubes (1.0, 1.5, 2.0 GHz). The INT-only reconstructions of the extended emission are clearly under-constrained.

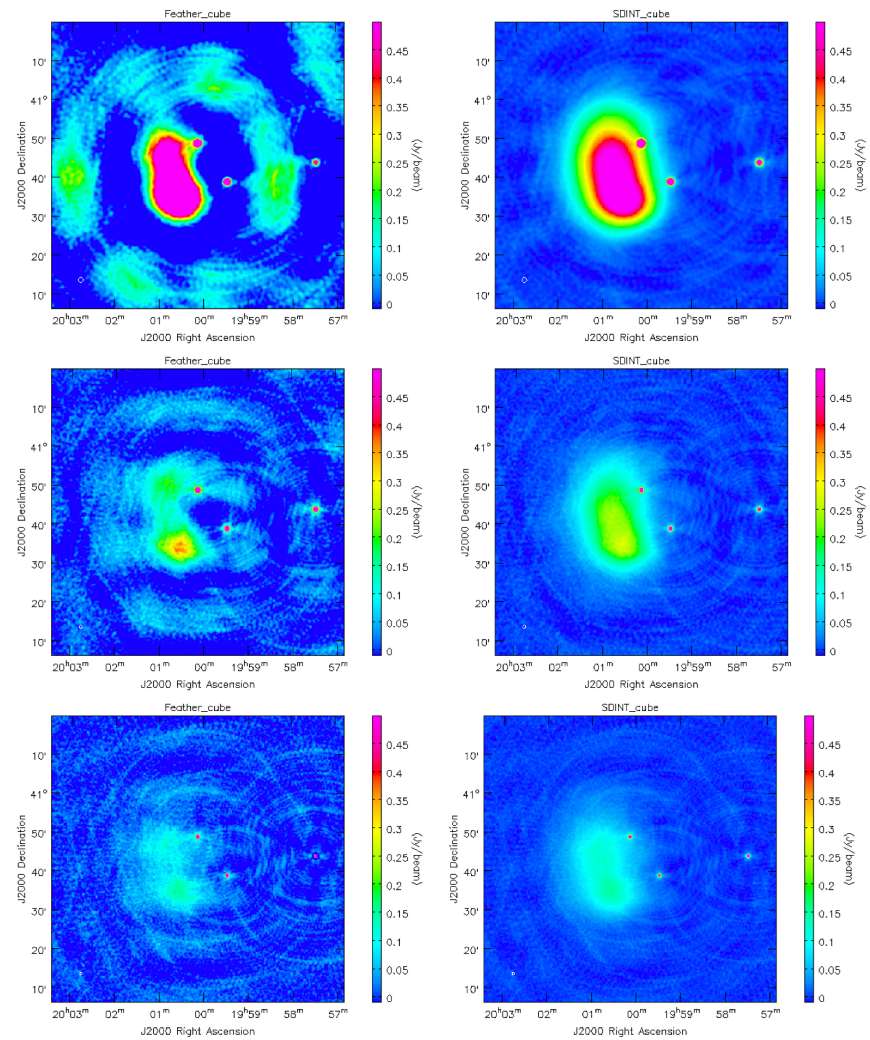


Figure 10. Feathered (LEFT) and Joint SDINT (RIGHT) Spectral Cube (1.0, 1.5, 2.0 GHz)

SD Data Scale in Feathering

- Same as feather SD scale factor, you can choose the ‘sdgain’ value in *runsdint.py*.
- Important if you notice large scale noise features from SD data corrupting the JD images.
- Reduce/increase ‘sdgain’ to down-weight/up-weight the SD data relative to the INT data.
- Too much change to ‘sdgain’ will bias the JD image toward INT or SD data.
- Check noise in INT-only, SD-only, and SDINT versions to see what values are needed for ‘sdgain’. Noise adds in quadrature here.
- Doesn’t change the integrated flux density since PSF is scaled by this factor too.

SDINT Status

- Currently being tested with real GBT+VLA data from CHANG-ES Survey (Irwin et al. 2012).
- Will use CHANG-ES data and M100 ALMA data as test cases.
- Released as a CASA task in few months to a year.