MeerKATHI

Developing an end-to-end data reduction pipeline for MeerKAT and other radio telescopes

The MeerKATHI collaboration

G. I. G. Józsa[1,2,3] (at this conference), S. V. White[2], K. Thorat[2,1,4], O. M. Smirnov[2,1], P. Serra[5], Mpati Ramatsoku[2,5], A. J. T. Ramaila[1], S. Perkins[1], D. Molnár[5], S. Makhathini[2], F. M. Maccagni[5], D. Kleiner[5], P. Kamphuis[6], B. V. Hugo[1,2], W. J. G. de Blok[7,8,9], L. A. L. Andati[2]

[1]South African Radio Astronomy Observatory (SARAO, South Africa)
[2]Department of Physics and Electronics, Rhodes University (South Africa)
[3]Argelander-Institut für Astronomie (AIfA, Germany)
[4]Department of Physics, University of Pretoria (South Africa)
[5]INAF - Osservatorio Astronomico di Cagliari (Italy)
[6]Ruhr-Universität Bochum, Faculty of Physics and Astronomy (AIRUB, Germany)
[7]Netherlands Institute for Radio Astronomy (ASTRON, The Netherlands)
[8]Dept. of Astronomy, University of Cape Town (South Africa)
[9]Kapteyn Astronomical Institute, University of Groningen (The Netherlands)

Synopsis

MeerKATHI is the current development name for a data reduction pipeline, assembled by an international initiative. We create a publicly available end-to-end continuum- and line imaging pipeline for MeerKAT and other radio telescopes. We implement high-end techniques suitable to produce high-dynamic-range continuum images and spectroscopic data cubes. Using containerization, our pipeline is platform-independent. Furthermore we are using a standardized approach to use a number of different of high-end software suites, partly developed within our group. We are aiming at using distributed computing approaches throughout our pipeline to enable the user reduce larger data sets as provided by radio telescopes like MeerKAT. The pipeline also delivers a set of imaging quality metrics giving the user the opportunity to efficiently access the data quality.

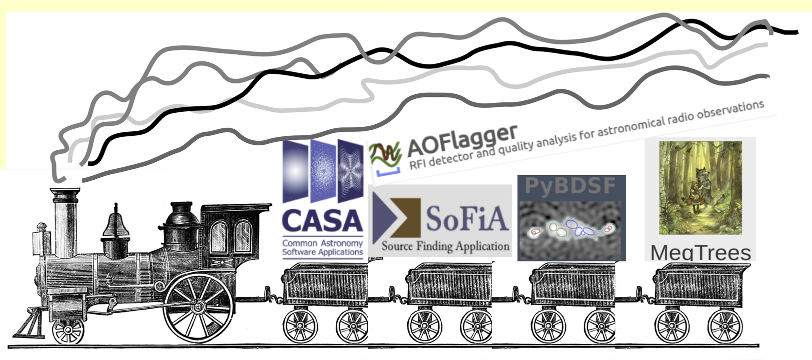
Python, Stimela, and with that, the world

MeerKATHI is a (large) Python script making use of Stimela (the IsiZulu word for a train), a platform-independent radio interferometry scripting framework based on Python and a choice of Linux containerization technologies. It enables the user to use a suite of astronomical software to reduce radio astronomical imaging data (as produced by telescopes like MeerKAT, the VLA, etc.). In addition to containerized and therefore platform-independent versions of the software it provides a standard syntax, allowing the user to access all implemented software in the same manner. Stimela supports the containerization platforms:

- Podman
- Docker
- Singularity
- uDocker

It provides access to standard software as CASA (McMullin et al. 2007, generic data reduction and analysis), PyBDSM (Mohan & Rafferty 2015, see also van Weeren et al. 2012, source finding), AOFlagger (Offringa 2010, radio interference detection and elimination), MeqTrees (Noordam & Smirnov 2010, data simulation and calibration), SoFiA (Serra et al. 2015, source finding). In addition, it wraps high-end software developed within our group, using the same standard syntax:

- CubiCal (Kenyon et al. 2018, prediction and calibration)
- AIMFAST (Diagnostics/flow control)
- RFInder (RFI visualisation)
- Radiopadre (Results examiner)
- RAGAVI (Data visualisation)
- Tricolour (Parallel flagger)
- Sunblocker (Solar RFI mitigation)
- Crystalball (Parallel predict)
- Sharpener (Spectral analysis)
- Codex Africanus (parallel radio software API)

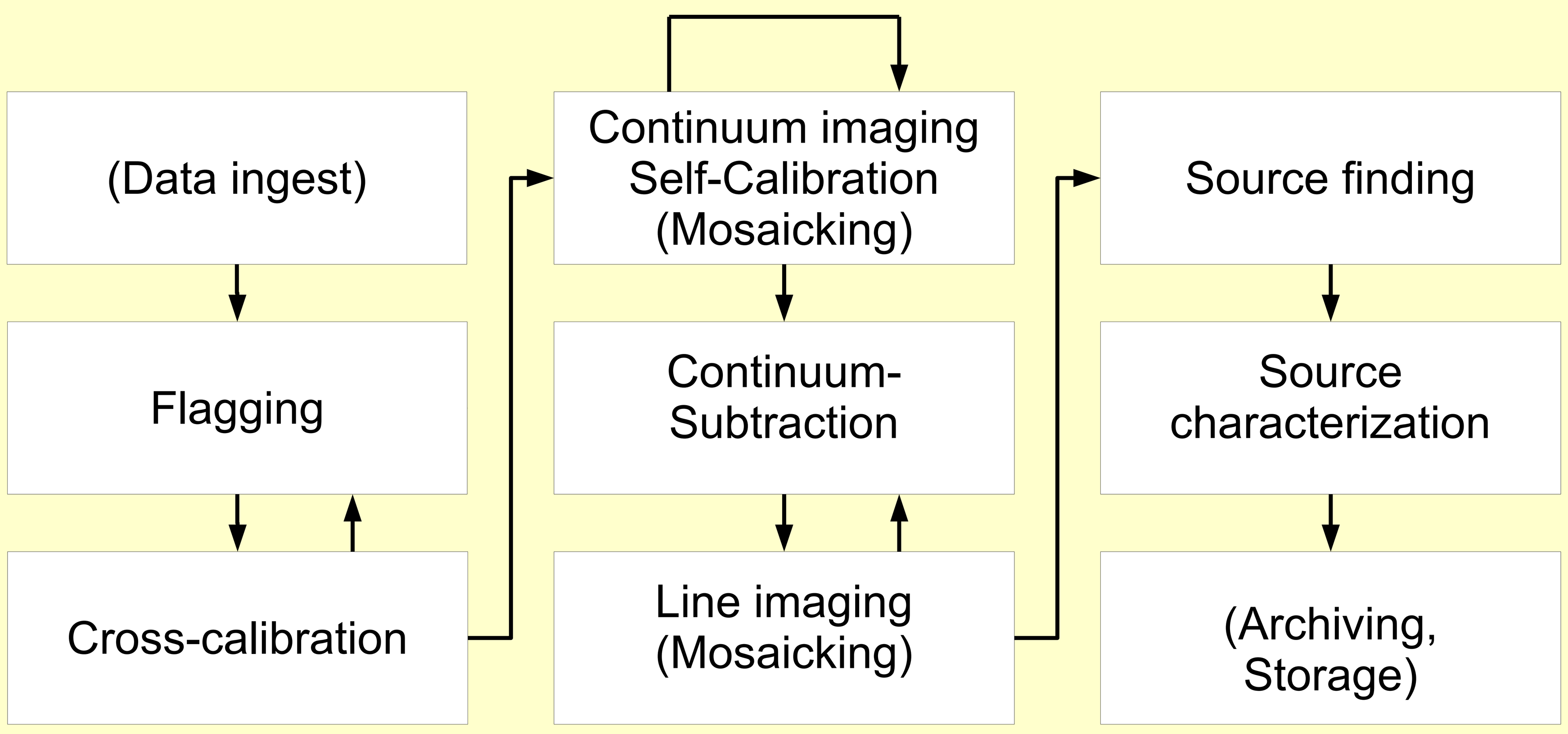


Stimela

Repositories

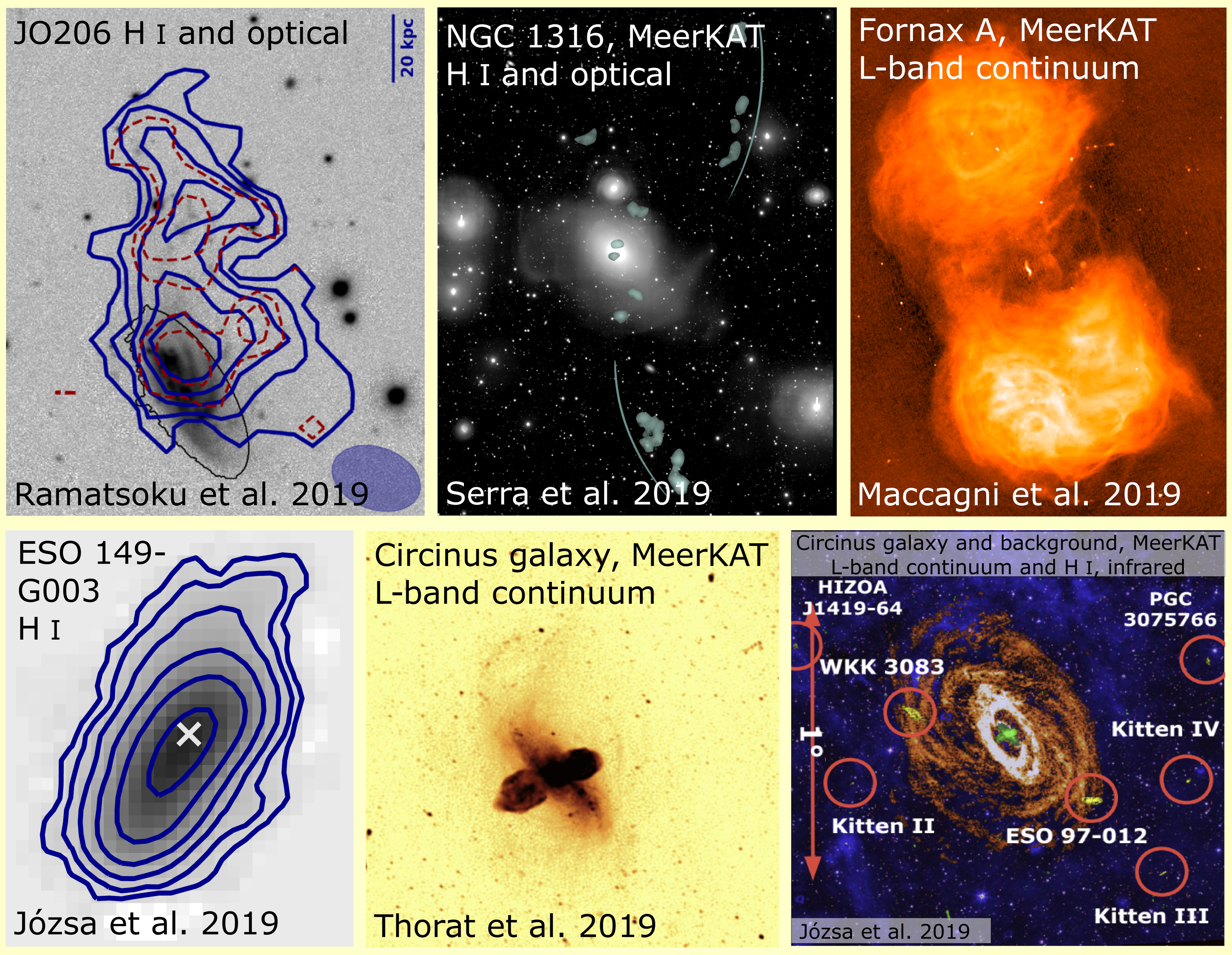
Podman:	https://podman.io/
Docker:	https://www.docker.com/
Singularity:	https://sylabs.io/singularity/
uDocker:	https://github.com/indigo-dc/udocker
CASA:	https://casa.nrao.edu/
PyBDSM:	https://github.com/lofar-astron/PyBDSF
AOFlagger:	https://sourceforge.net/p/aoflagger/wiki/Home/
MeqTrees:	http://meqtrees.net/
SoFiA:	https://github.com/SoFiA-Admin/SoFiA
CUBIcal:	https://github.com/ratt-ru/CubiCal
AImFAsT:	https://github.com/Athanaseus/aimfast
RFInder:	https://github.com/Fil8/RFInder
RadioPADRE:	https://github.com/ratt-ru/radiopadre
RAGaVi:	https://github.com/ratt-ru/ragavi
Tricolour:	https://github.com/ska-sa/tricolour
Sunblocker :	https://github.com/gigjozsa/sunblocker
Crystalball:	https://github.com/paoloserra/crystalball
SHARPener:	https://github.com/Fil8/SHARPener
Codex Africanus:	https://github.com/ska-sa/codex-africanus

Work flow



MeerKATHI has a flexible layout and gets configured via a configuration file. Its simplest function is to flag RFI, to perform a cross-calibration (using calibration sources), followed by an iterative series of continuum imaging and self-calibration steps (reducing image artifacts based on the structure of output images), line imaging, and source finding. Automated data ingest (depending on the telescope), several stages of source characterization, and archiving are planned, but not yet implemented.
A suite of fast, interactive, remotely accessible diagnostic tools has been developed to access data quality and to identify potential problems.

First results and outlook



MeerKATHI is under development since May 2017. Since then it has been tested using several data sets from MeerKAT, including commissioning, and open time data (Serra et al. 2019, Maccagni et al. 2019, Józsa et al. 2019a,b, Thorat et al. 2019), but it has also been used to image VLA data (Ramatsoku et al. 2019). It has been used successfully to image the H I (neutral hydrogen) line (Ramatsoku et al. 2019, Serra et al. 2019, Józsa et al. 2019a,b) as well as radio continuum data (Maccagni et al. 2019, Thorat et al. 2019).
The pipeline is being ramped up and documented to be publicly released this year, while development will continue.

Literature

Józsa, G. I. G., et al. 2019a, submitted to MNRAS
Józsa, G. I. G., et al. 2019b, in prep.
Kenyon, J. S., Smirnov, O. M., Grobler, T. L., Perkins, S. J. 2018, MNRAS, 478, 2399
McMullin, J. P., et al. 2007, Astronomical Data Analysis Software and Systems XVI (ASP Conf. Ser. 376), ed. R. A. Shaw, F. Hill, & D. J. Bell (San Francisco, CA: ASP), 127
Mohan, N., Rafferty, D., ascl:1502.007
Noordam, J. E., Smirnov, O. M. 2010, A&A, 524, A61
Offringa, A. R., 2010, AOFlagger: RFI Software (ascl:1010.017)
Ramatsoku et al. 2019, MNRAS, 487, 4580
Serra, P. , et al. 2015, MNRAS, 448, 1922
Serral, P. et al. 2019, A&A, 628, A122
van Weeren, R. J., et al. 2012, A&A, 543A, 43

https://meerkathi.readthedocs.io

meerkathi@googlegroups.com