Project 343

M.A. Brentjens

November 11, 2005

1 Goal

The goal of this project is to boost the MeqTree functionality to a point where it can actually be used for regular datareduction.

2 Necessary steps

- Preprocess datasets with Aips++ : Flagging, Tsys, bandpass
- Repeat bandpass calibration with MeqTree system and compare
- Do full phase and amplitude calibration on main dataset

3 T_{sys} correction

There is no T_{sys} correction functionality present in Aips++. Therefore I have written my own Glish tool to apply system temperature corrections. This is a pure necessity in the case of the WSRT. All visibilities are multiplied with $\sqrt{T_{1i}T_{2j}}$, where i and j are the respective polarizations, e.g. x and y. If a specific system temperature is not available, we take the one from the previous time slot. If that one is also invalid or not avaulable, we take the average value over all valid system temperature readings of the same time slot. A system temperature reading is invalid if it is $\pm$`inf`, 0, or `NaN`.

The class can only handle full-polarization data. It is assumed that the output column exists.

xx and yy amplitude of an unpolarized source should be I/2. Instead in Aips++, it is assumed to be I. I counter this by multiplying the MODEL_DATA column by 0.5 after having called imgr.setjy(). A consequence is that before imaging using the Aips++ imager, we should multiply the CORRECTED_DATA column by 2 in order to get a proper fluxscale in the image plane.

4 Expected image noise per channel

12h run, 156250 Hz bandwidth per channel. WSRT calculator Stokes I image noise: 0.3 mJy/beam per channel

5 Flagging

The autoflagger happily flags correct data on steep fringe flanks. Partial "solution": flag only on XY and YX with high threshold ("20").

6 Sources

The dominant sources are:

Name	RA	Dec	Flux1175 MHz
3C 343.1	16 38 28.2	+63 34 44	5.0
3C 343	16 34 33.8	+62 45 36	5.2

Channels:

1	10	20	30	33	40	50	60	64
1180.00	1178.59	1177.03	1175.47	1175.00	1173.91	1172.19	1170.63	1170.16

Channel width: 156250 Hz

7 Solver

Made solver aware of flags: doesn't add flagged equations to the solver anymore

Use another makeNorm() method, that copes with sparse equations

8 UVW

MeqUVW UVWs do not correspond to MS UVWs...

The solution is simple: when retrieving ANY dataset from the WSRT archive, one should first run j2convert. It computes J2000 UVWs from the station XYZ coordinates and the UTC date:

j2convert msin=¡MSDIRNAME¿

9 Spigot/Sink

Freq/time exchanged

10 343

Phase centre: 16:38:28.205 +62:34:44.314 4.3566476092072053 1.0922091756657659

Solve fails on first timeslot. Solve succeeds for subsequent timeslots, but result is dubious at the moment.

It turned out the positions in solve343.g were wrong: The correct ones are:

J2000.0: 3C343.1: 16h38m28.180s +62d34m44.16s 4.356645791155902 1.092208429052697 lm: (-8.3726023633066311e-07, -7.4661239335416241e-07) n-1: (-6.2927441035753873e-13)

J2000.0: 3C343: 16h34m33.789s +62d45m35.81s 4.3396003966265599 1.0953677174056471 lm: (-0.0078024647942417169, 0.0032175709681556697) n-1: (-3.5616244158709343e-05)

Positions: Kühr et al 1981. No VLBI positions known...

src_ra := ([4.356645791155902,4.3396003966265599]); src_dec := ([1.092208429052697,1.0953677174056471
src_sti := [1,1];

The computed visibilities of MeqStatPointSourceDFT vary WAY too rapidly...

It's the "Wwwwwww" term !?!?!... Figure this out over the weekend...

W term issue solved. Correct visibility for a point source is

$$V_{\rm xx} = \frac{1}{2n} I \mathrm{e}^{2\pi \mathrm{i}(ul+vm+w(n-1))\nu/c} \qquad (1)$$

The "-1" in $(n-1)$ is due to the fringe stopping in the array.

The divisioon by n comes from the integral over a solid angle.

The source fluxes found after 10 iterations on a dataset that is not yet phase-calibrated are are:

3C343.1: 5.38604422187 Jy 3C343 : 1.63316710711 Jy

These are found on a dataset that DID NOT HAVE Tsys corrections applied...

The solution converged to 7 significant numbers in 2 iterations. After 1 iteration, the solution was:

3C343.1: 5.38166154461 Jy 3C343: 1.63252761647 Jy

3C 343 flux corresponds to a true flux of 6 Jy after primary beam correction

11 Needs:

WSRTPrimaryBeam Node.

$$P = \cos^6(k(\nu)\nu r), \qquad (2)$$

where $k(\nu) \approx 0.0651934 + 2.68503 \times 10^{-6}\nu - 6.25456 \times 10^{-10}\nu^2$, where ν is in MHz and r is the angular distance between the pointingcentre and the source.

The inputs of the node should be: ra_pointing, dec_pointing, ra, dec in radians.

The output is P computed for every value in the domain.

The angular distance between (ra_pointing, dec_pointing) and (ra,dec) may be computed via (pseudocode):

```
a = abs(ra - ra_pointing)
B = abs(0.5*pi - dec_pointing)
C = abs(0.5*pi - dec)
arg = cos(B)*cos(C) +sin(B)*sin(C)*cos(a)

if arg >= 1.0 then arg = 1.0
if arg <= -1.0 the arg = -1.0
angular_distance = arccos(arg)
```

Spectral dependence of source flux

Saving parameters.

12 A scale factor

Tue Apr 5 17:25:51 CEST 2005

There appears to be a scale factor of 40 between the Newstar phases for telescope B.:

Gnuplot: plot 'RTB.txt' using :((2+0.021)∗39)$withlines,' 343_{n}ewstar_{x}phases'using$((1+90)*((23+56/60.0
with lines 3

The scale factor appears to arise from an optimization that is spread over PointSourceDFT and StatPointSourceDFT.

Fri Apr 8 17:06:14 CEST 2005: implemented new "VisPhaseShift" class that computes

$$\mathrm{e}^{-2\pi \mathrm{i}\mathbf{u}\cdot\mathbf{l}\nu/c} \tag{3}$$

updated the solve343.g script to reflect the changes necessary to incorporate the VisPhaseShift node. For example: the VisPhaseShift node is fed (u,v,w) and (l,m,(n-1)).

13 Matrix measurement Equation

Tue Apr 19 15:35:13 CEST 2005

$$V_{ij} = \sum_{k=1}^{N} G_i J_{ik} K_{ik} S_k K_{jk}^{*} J_{jk}^{\dagger} G_j^{\dagger} \tag{4}$$

When solving for stokes I, meqserver runs out of memory.

Wed May 18 09:07:41 CEST 2005

Memory leak solved by Oleg. There still remains a small memory leak in ParmtTable. Memory consumption is about 150 MB after 370 timeslots, 16 channels, and slowly increasing. I've switched off writing to an MS and now it takes 4.4 seconds per timeslot. Oleg says that converting the MS to BOIO in advance, could speed up meqserver to do about one timeslot per second. I have not yet verified that, bu will do this in the course of the day.

Wed May 18 13:32:34 CEST 2005

Memory usage for source flux fit: 2 / 4 parameters, 16 channels, 1437 timeslots: 560 MB at start of iteration, 1430 at end. Every time, memory is destroyed at end of iteration and re-allocated during the iteration. This may be a large fraction of the overhead.

458 MB stable after computations are completed.

Thu May 26 09:51:55 CEST 2005

created maps. Channels 25,26,27,28 and 37,38,39,40 have increased noise. ALSO IN PREPROCESSED DATA COLUMN!

14 Processing in MeqTree

- preprocess data (crosscal, preliminary flagging
- source flux fit (I+Q)
- common phase solution (30s)
- source flux fit (I+Q)
- common phase solution (30s)

- independent gain solution (15 min)

make cleaned images of central 8 channels (29,30,31,32,33,34,35,36) counting from 1. Make MFS image of those 8 channels.

Clark CLEAN downto 1 mJy in max 5000 iterations (padding 1.5)

2048 squared pixels of 4 arcsec diameter. Only inner quarter is cleaned and most of the disturbing sources reside outside inner 512 pixels.

15 full automation

find footer:

mqs.relay gives access to agent wait on data_set_footer event.

After phase selfcal: flag_column_abs('3C343.MS', 'CORRECTED_DATA', 0.57)

16 Moving to TDL

16.1 Source model

A pointsource is a python class with the following fields:

Field	Contents
name	Name of source
ra	Right ascension in J2000.0 radians
dec	Declination in J2000 radians
I, Q, U, V	Stokes fluxes in Jy. If value is `None` the default value should be read from the source MEP table
table	string: MEP table name for the source
Iorder etc.	Order of I polc. Similar for Q, U, and V

16.2 Solver settings/tiling/solvables

16.3 Node naming conventions

If a node is specific to an antenna, it ends with the (0-based) antenna number. If a node is specific for a baseline ANT1-ANT2, it ends with <ANT1-number>.<ANT2-number>.

16.4 The tree

The main branch is the predict branch, which contains the measurement equation:

$$V_{ij} = \sum_{k=1}^{N} G_i J_{ik} K_{ik} S_k K^*_{jk} J^\dagger_{jk} G^\dagger_j \quad (5)$$

$$S_k = \begin{pmatrix} xx & xy \\ yx & yy \end{pmatrix} /n \quad (6)$$

$$xx = fracI + Q2 \quad (7)$$

$$
\begin{aligned}
xy &= fracU - \mathrm{i}V2 && (8)\\
yx &= fracU + \mathrm{i}V2 && (9)\\
yy &= fracI - Q2 && (10)\\
&&& (11)
\end{aligned}
$$

16.5 Global Nodes

Name	Function
ra0	RA of field centre in J2000 radians
dec0	DEC of field centre in J2000 radians
one	1.0: MeqConstant
half	0.5: MeqConstant

16.6 Source Nodes

- RA.<sourcename>
- DEC.<sourcename>
- LMN.<sourcename>
- L.<sourcename>
- M.<sourcename>
- N.<sourcename>
- I.<sourcename>
- Q.<sourcename>
- U.<sourcename>
- V.<sourcename>
- IQUV.<sourcename>
- Coherency.<sourcename>
- Coherency.<sourcename>