

Planck-toi-HiRes Documentation

Eric J. Murphy

1 Introduction

HiRes is a tool that allows one to reassemble the information from overlapping observations into a single image with refined resolution, justified by the increased spatial frequency coverage provided by the redundancy, in conjunction with reliable knowledge of the point spread functions. The algorithm is based upon the Maximum Correlation Method (MCM) developed at the Caltech Infrared Analysis Center (IPAC) for use with data from the Infrared Astronomical Satellite (IRAS; see Aumann et al. 1990). MCM is a generalization of the Richardson-Lucy (R-L) algorithm (Richardson 1974; Lucy 1972) to redundant coverage. A detailed discussion of HiRes usage on *Spitzer* data can be found in Backus et al. (2005).

2 Usage with *Planck* Data

The routine *planck_toi_hires* (**v1.0.0; 20151109**) reads in a data file with *Planck* TOI data points (i.e., those generated by the IRSA/*Planck* TOI server) and creates FITS images out of them. The installation procedure and an example on how to fetch *Planck* TOI data from IRSA is provided in the “README” file included with the source download package. Additionally, a list of dependencies along with an example of a simple invocation of the code on such a TOI data file is provided in the “planck_toi_utils/doc/README.hires” file included with the source package. Below we provide a summary of the information in these README files on how to both retrieve *Planck* TOI data from IRSA and run *planck_toi_hires* to create HiRes and Minimap images.

Planck TOI data can be obtained interactively via the IRSA/*Planck* webpage, or at the command line with a command such as:

```
curl -o planck_toi.fits 'http://irsa.ipac.caltech.edu/TAP/sync?QUERY=
SELECT**FROM+planck_toi_030+WHERE+CONTAINS(POINT
('J2000',ra,dec),CIRCLE('J2000',121.17440,-21.57294,0.1))=1&format=fits"
```

In this example, which is for a dataset at 30 GHz, the code can be called from the command line:

```
./planck_toi_hires --query='input_file:'test/hires_minimal.fits',
detector:30, mcm_iterations:32, output:prefix:'test/hires_minimal/',
type:['minimap',"mcm"], pos:angResolution:0.01"
```

Alternatively, one can use a parameter file to run *planck_toi_hires*:

```
./planck_toi_hires test/hires_minimal.json5
```

Either invocation will generate a file called “minimap.fits” that should be identical to the one generated by the online Minimap image service along with a file called “mcm.fits”, which for this case would be a HiRes deconvolution run for 32 iterations. The control file format is JSON5, an easier to read and write version of JSON. There are two examples provided in the “planck_utils/test” directory included in the downloaded source package. The complete list of parameters and their meaning is detailed in the schema provided here: “planck_toi_utils/doc/planck_toi_hires.schema.json” in the source package and here: “input/planck_toi_hires.schema.json” in the binary package.

3 Examples

To characterize the performance and limitations of the deconvolution algorithm, we carried out two fairly representative examples. The first example was for a field having a compact source with a low background (Centaurus A). The second example was for a field having an extended source with an elevated background (30 Doradus). We ran the MCM algorithm for 2, 5, 10, and 20 iterations at each *Planck* frequency, comparing image characteristics of each deconvolution with what was obtained by the Minimap image reconstruction. The resulting images are shown in Figure 1 for Cen A and in Figure 2 for 30 Dor.

We additionally fit 2-D Gaussians to each image to measure how much the resolution is in fact improved over the Minimap image reconstruction. The major and minor axis values from the fits are given in Tables 1 and 2

for Cen A and 30 Dor, respectively. We additionally characterize the image fidelity by measuring the Strehl ratio, which we define as the ratio of the total flux measured within an aperture having a diameter equal to the Gaussian fit major axis for the MCM deconvolution relative to the same value measured for Minimap. Thus, a ratio of unity suggests that there is no flux loss introduced by the deconvolution. The Strehl ratios are given in Tables 3 and 4 for Cen A and 30 Dor, respectively.

Table 1: 2-D Gaussian Fit FWHM Values for Cen A

Frequency (GHz)	Minimap	MCM (2-iter)	MCM (5-iter)	MCM (10-iter)	MCM (20-iter)
30	30'71 × 38'71	33'02 × 40'34	22'26 × 29'66	16'03 × 24'43	12'89 × 19'57
44	28'47 × 29'53	16'48 × NaN	10'69 × 12'72	15'22 × 16'48	6'79 × 3'65
70	13'21 × 14'86	13'53 × 13'88	7'70 × 10'47	1'96 × 2'42	2'83 × 1'00
143	8'25 × 8'66	8'37 × 8'63	6'23 × 6'49	1'82 × 1'71	1'03 × 2'07
217	5'72 × 6'12	6'11 × 6'45	4'67 × 5'01	3'94 × 4'25	3'33 × 3'59
353	5'51 × 5'60	5'96 × 5'98	4'59 × 4'64	3'89 × 3'95	3'37 × 3'43
545	5'58 × 5'26	6'09 × 5'72	4'76 × 4'46	4'05 × 3'80	3'53 × 3'32
857	5'20 × 5'18	5'67 × 5'50	4'40 × 4'30	3'73 × 3'70	3'25 × 3'28

Note: Entries having NaN's indicate that the fit failed to converge.

Table 2: 2-D Gaussian Fit FWHM Values for 30 Dor

Frequency (GHz)	Minimap	MCM (2-iter)	MCM (5-iter)	MCM (10-iter)	MCM (20-iter)
30	37'41 × 41'38	39'11 × 42'93	29'61 × 33'11	24'89 × 28'09	21'60 × 23'70
44	31'27 × 33'88	33'61 × 37'33	25'53 × 28'18	21'62 × 22'44	17'79 × 17'98
70	16'61 × 16'57	17'76 × 17'75	13'22 × 12'96	10'97 × 10'82	9'31 × 9'37
143	11'06 × 11'38	11'46 × 11'63	8'61 × 9'02	7'39 × 7'85	6'60 × 7'04
217	10'55 × 11'77	10'26 × 11'37	7'79 × 8'96	6'43 × 7'54	5'32 × 6'52
353	15'32 × 16'21	14'28 × 15'29	11'66 × 13'62	11'24 × 13'34	10'88 × 13'13
545	17'43 × 17'47	15'82 × 16'31	13'20 × 14'89	13'04 × 14'80	12'89 × 14'74
857	15'40 × 16'61	14'39 × 15'72	12'30 × 14'55	11'83 × 14'34	11'56 × 14'24

3.1 Results

Based on running the MCM deconvolution for our two examples, we find the following. First, it appears that the algorithm introduces artifacts at the

Table 3: Strehl Ratios for Deconvolutions of Cen A

Frequency (GHz)	MCM (2-iter)	MCM (5-iter)	MCM (10-iter)	MCM (20-iter)
30	1.01	0.96	0.94	0.89
44	0.59	0.59	0.87	0.50
70	0.90	0.98	0.20	0.37
143	0.99	0.98	0.26	0.51
217	1.05	0.99	0.94	0.86
353	1.09	1.00	0.93	0.88
545	1.13	0.99	0.92	0.86
857	1.11	1.00	0.93	0.88

Note: In all cases, the Strehl ratio is the measured power within an aperture having a diameter equal to the major axis given by the 2-D Gaussian fit in Table 1 for each deconvolution relative to what is obtained using Minimap.

Table 4: Strehl Ratios for Deconvolutions of 30 Dor

Frequency (GHz)	MCM (2-iter)	MCM (5-iter)	MCM (10-iter)	MCM (20-iter)
30	1.04	1.01	0.96	0.89
44	1.11	1.02	0.92	0.80
70	1.09	0.99	0.93	0.88
143	1.06	0.92	0.84	0.78
217	0.99	0.79	0.66	0.57
353	0.94	0.82	0.80	0.78
545	0.92	0.82	0.82	0.81
857	0.94	0.86	0.84	0.83

Note: In all cases, the Strehl ratio is the measured power within an aperture having a diameter equal to the major axis given by the 2-D Gaussian fit in Table 2 for each deconvolution relative to what is obtained using Minimap.

lowest *Planck* frequencies. Such artifacts become quite obvious in the image reconstructions at frequencies of 143 GHz and below once the MCM algorithm is run for 10 or more iterations. This appears to be especially true for the cases where the mean background level is ≈ 0 (e.g., the low frequency cases of Cen A). Such artifacts do not seem as commonly present for the image reconstructions where the mean background is elevated above 0, which is typically the case at the highest *Planck* frequencies. Thus, the deconvolution appears to generally be robust at frequencies of ≥ 217 GHz (i.e., when the background level is elevated) and problematic at the lower frequencies (i.e.,

where the mean background level is ≈ 0).

To quantify the resolution improvement by the MCM algorithm, we use the compact source example of Cen A, focussing on the frequencies that did not have obvious artifacts introduced by the deconvolution (i.e., ≥ 217 GHz). The FWHM measured by the Minimap image of Cen A at these frequencies is roughly 20, 40, and 60% larger than the MCM deconvolution after running the algorithm for 5, 10, and 20 iterations, respectively. Consequently, the improvement in resolution is already beginning to flatten out by increasing the iteration number beyond 10 iterations.

Based on these findings, we come to the following suggestions for the usage of MCM on *Planck* data. MCM appears to generate fairly robust images with intensity conservation when using up to 10 iterations at frequencies ≥ 217 GHz (i.e., where the background levels are elevated above 0), and thus we recommend this as a conservative guideline. Of course, there appears to low frequency cases where the MCM algorithm did not introduce any obvious artifacts, as well as increased resolution improvement by using up to 20 iterations based on our examples. Thus, depending on the source and the nature of the background, individual users might find convergent solutions beyond 10 iterations, but we recommend extreme caution when interpreting those results.

References

- Aumann, H. H., Fowler, J. W., & Melnyk, M. 1990, *Astronomical Journal*, 99, 1674
- Backus, C., Velusamy, T., Thompson, T., & Arballo, J. 2005, *Astronomical Data Analysis Software and Systems XIV*, 347, 61
- Lucy, L. B. 1974, *Astronomical Journal*, 79, 745 Richardson, W. H. 1972, *Journal of the Optical Society of America* (1917-1983), 62, 55

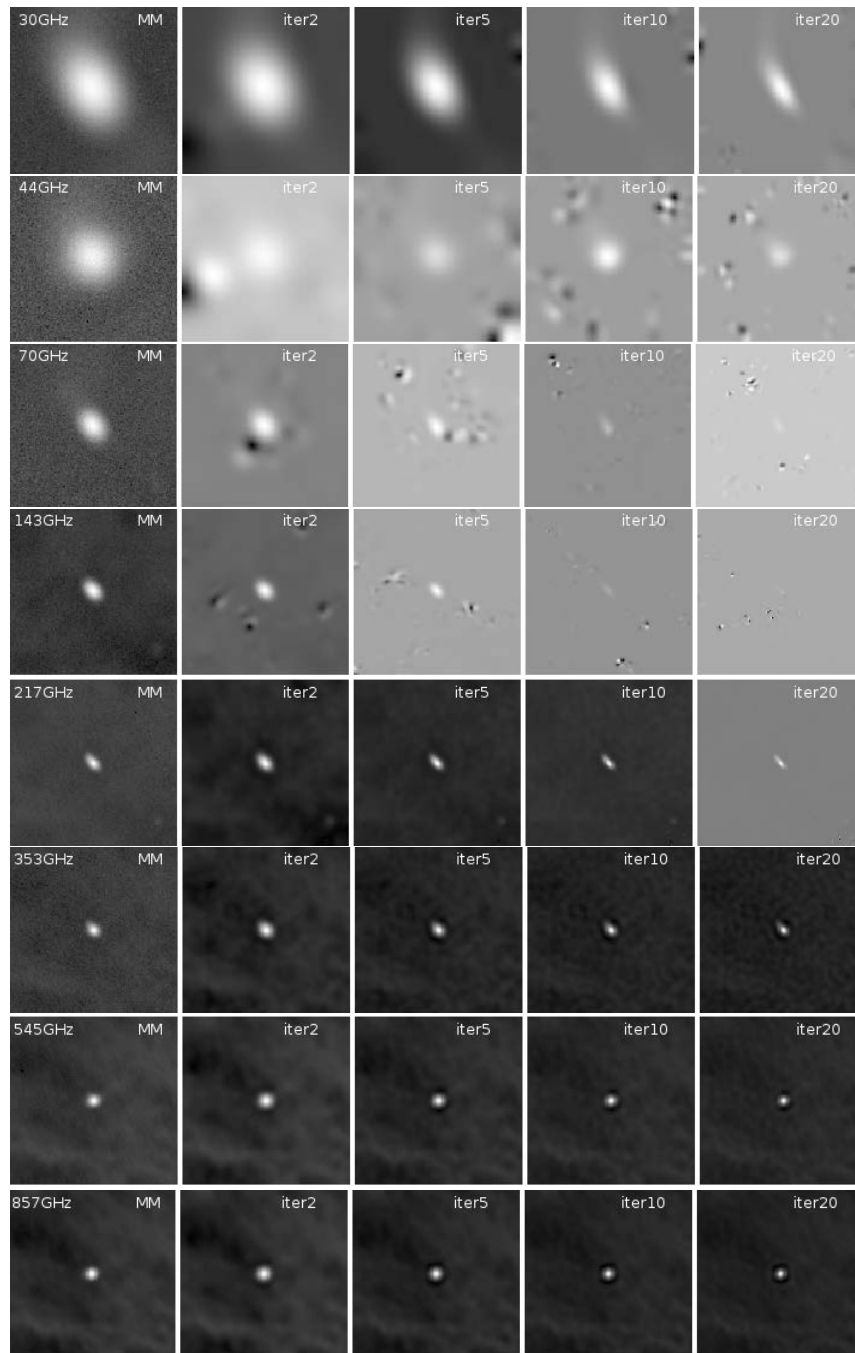


Figure 1: Images of Centaurus A ($2^\circ \times 2^\circ$) at each *Planck* frequency (per row). The first column shows the Minimap image construction, followed by MCM deconvolutions run for 2, 5, 10, and 20 iterations from left to right, respectively. Image reconstruction at 30 – 143 GHz seem to show significant artifacts at all iterations. This is likely a result of the MCM algorithm finding mean ≈ 0 level backgrounds challenging to converge on.

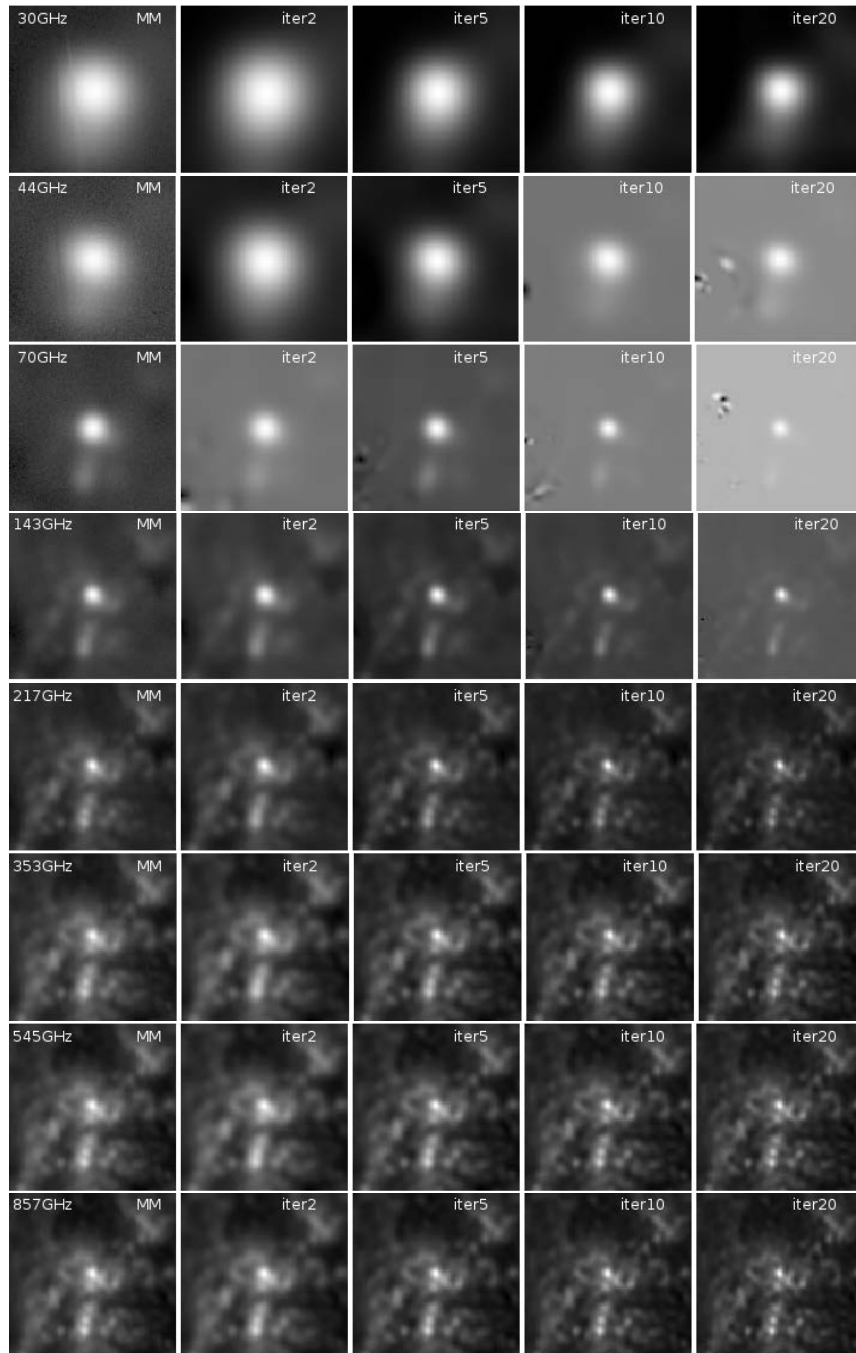


Figure 2: Images of 30 Doradus ($2^\circ \times 2^\circ$) at each *Planck* frequency (per row). The first column shows the Minimap image construction, followed by MCM deconvolutions run for 2, 5, 10, and 20 iterations from left to right, respectively. For bright sources on elevated backgrounds, we find high fidelity images at 143 GHz and higher, for up to 20 iterations.