Progress Report #13.1 (December 2010)

I repeated the same simulations as before (2-source sky model, each source 1 Jy in brightness and separted by ~1 FHWM ~ 75 arcmin, SKA log-spiral array with N_a = 50, 75, 100, 150, 175, without visibilities corrputed by the cortes beam model, with the visibilities corrupted by the cortes beam model with and without the pointing error model applied, difference FITS images, etc.) except in this case I generated the empty MS with CASA and used CASA as well to generate the image maps. As before, Meqtrees was used to implement the 2-source sky model and to corrupt the visibilities with the cortes beam and pointing error models. I also obtained statistical measures of a region between and away from the two sources in each of the images and generated corresponding line plots of counts vs. pixel brightness values. These simulations might serve as a "sanity check" or double-checking of the previous simulations where Meqtrees was used to (also) generate the empty MS's and the image maps, as well as a launching point for experimenting with the A-projection algorithm or similar algorithms to try to correct for or reverse the effects of the E-Jones / Cortes beam model.

The results (see the attached file
"figures_plots_logspiral50_175_casa_2src_75arcmin_poi1_imagediff.pdf",
which includes self-explanatory captions and labeling of images and line plots) seem to be broadly consistent with the results of the previous simulations as reported in previous progress reports, though I have not yet generated summary tables of statistical results to better confirm this (-- see progress report #13.2 within the next couple of days). Once again, source #2 is greatly suppressed (to the point where it is essentially invisible in some cases) for those images in which the visibilities were corrupted with the cortes beam model (with and without pointing errors). Also, for these simulations, there seems to be a clearer trend indicating that source #2 is effected more by the cortes beam model and the pointing errors than source #1, as we would expect. For example, difference images between images where visibilities where corrupted with the cortes beam model and pointing errors and images with visibilities corrupted by the cortes beam model without pointing errors show source #2 to appear brighter than source #1. This (qualitative) trend was not as clear or strong as in the previous simulations where Meqtrees was used to generate the empty measurement sets and the lwimager was used to generate image maps.

Progress Report #13.2 (December 2010)

This progress report is a follow-up, supplement, and complement to Progress Report #13.1
that I sent you a couple of days ago (and which is forwarded below for reference).

I've generated summary tables of statistical measures (attached to this e-mail) for the simulated images that I discussed and sent you in Progress Report #13.1. Remember, for these latest simulations, I used CASA to run SKA simulated observations and generate corresponding empty measurement sets, then used Meqtrees to implement the 2-source sky model and to corrupt the visibilities with the Cortes beam model and pointing errors, then used CASA again to make dirty images/maps (using w-projection as well)from the simulated visibilities.

Broadly speaking, the statistical results seem to be generally consistent with what we can see at a more qualitative level with the image maps that I sent you and which were discussed in Progress Report #13.1, and also broadly consistent with the previous progress reports for the previous simulations.

The two tables include (hopefully) self-exaplanatory captions, which also describe the different labelings used in the left-most (1st) columns of each table, for example.

Table 1 gives statistical measures within a bounding box region drawn between but not including the two sources (in columns 2 through 5). Column 6 of Table 1 gives a dynamic range estimate for the entire image, and Column 7 gives the pixel range (min,max) for the entire image.

Table 2 includes statistical measures for the two sources themselves, i.e., within two small bounding box regions drawn around each source. Note that source #1 (=src1) signifies the source at field center, and source #2 (=src2) signifies the source separated from field center by ~ 1 FHWM ~ 75 arcmin.

The following are some of the clearest and most persistent trends I was able to find among the statistical measures in each of the tables:

(1) For those images created from visibilities that were *not* corrupted by the Cortes beam model, the dynamic range estimates increased as N_a increased.

(2) For those images created from visibilities that *were* corrupted by the Cortes beam model, the dynamic range estimates also increased as N_a increased.

(3) For each N_a, the dynamic range estimates *increased* for images created from visibilities corrupted by the Cortes beam model as compared to images created from visibilities not corrupted by the Cortes beam pattern model.

(4) For each N_a, the standard deviation about the mean of pixel brightness values in a bounding box region between and away from the two sources was *smaller* in images generated from visibilities corrupted by the Cortes beam pattern model than in images generated from visibilities *not* corrupted by the Cortes beam pattern model.

(5) For images created from visibilities not corrupted by the Cortes beam model, the standard deviation about the mean of pixel brightness values in a bounding box region between and away from the two

sources generally *decreased* as N_a increased.

(6) For images created from visibilities that *were* corrupted by the Cortes beam model, the standard deviation about the mean of pixel brightness values in a bounding box region between and away from the two sources generally also *decreased* as N_a increased.

(7) The means and medians for pixel brightness values in the bounding box regions drawn around source #2 were significantly *smaller* (and also smaller compared to source #1) in images generated from visibilities corrupted by the Cortes beam model as compared to images generated from visibilities not corrupted by the Cortes beam model. This is consistent with our earlier qualitative assessments from visual examinations of the images (Progress Report #13.1 and previous progress reports) that source #2 becomes significantly suppressed compared to source #1 in images generated from visibilities corrupted by the Cortes beam model.

(8) For all images generated from visibilities corrupted by the Cortes beam model (with and without pointing errors applied and for all N_a), the standard deviation about the mean of pixel brightness values within the bounding box region drawn around source #2 was *smaller* than the standard deviation about the mean of pixel brightness values within the bounding box region drawn around source #1.

(9) For all images generated from visibilities corrupted by the Cortes beam model (with and without pointing errors applied and for all N_a), the root mean square of pixel brightness values within the bounding box region drawn around source #2 was *smaller* than the root mean square of pixel brightness values within the bounding box region drawn around source #1.

(10) In at least some of the statistical results for the differenced images, the mean of pixel brightness values within the bounding box region drawn around source #2 is *larger* that the mean of pixel brightness values within the bounding box region drawn around source #1. This seems to be consistent with our qualitative assessment from an examination of images from the last progress report (#13.1) that the effect of the cortes beam model and the pointing errors (i.e., source suppression) increases as one gets further from the pointing center (e.g., there is greater suppression of source #2 than source #1).

NOTE: In many cases, the statistical measures (e.g., means, medians, etc.) were negative values because many of the pixel brightness values are negative. So, for example, in some of the difference images we are subtracting a negative pixel value from another negative pixel value, or a positive pixel value from a negative one, etc. leading to negative means and medians in the differenced images also, which can be difficult to interpret and which should be evaluated cautiously and carefully.

Addendum/Appendix to Progress Reports #13.1 and #13.2 (December 2010)

1. As an addendum and appendix to Progress Reports #13.1 and #13.2, I just wanted to send you (attached), just in case this is useful and/or of interest to you, the CASA python scripts I used (along with Meqtrees) in carrying out the simulations of SKA observations of the 2-point sky source model and in obtaining statistical measures of the simulated images:

* SKA_VLA_simulator_casa_2src_read.py: This is the CASA python script I used to create the UV tracks for SKA simulations and to create the empty measurement set. This script includes a new useful feature that allows one to read in the radio antenna array position lay-out from a text file (e.g., "ants_Na150_logspiral1a.txt", also attached to this e-mail).

* CASA_Imager_SKA_2src.py: This is the CASA Python script I used to form an image map from the visibilities that were corrupted by the cortes beam model (as implemented in the Meqtrees simulations).

* CASA_plot_stats_2src.py: This is the CASA Python script I used to obtain statistical measures (e.g., pixel brightness value means, medians, root mean squares, standard deviation about the mean, etc.).

2. I downloaded and successfully installed on the myska7 VM the latest stable version of CASA for Linux (CASA release 3.1.0), which was compiled on 11/30/2010. I've subsequently learned that the previous version of CASA I was using and had installed on the VM (Version 3.02) *did* in fact have a working and apparently usable 'aprojection' algorithm function. I had not correctly implemented it in the CASA commands I had used previously, leading me to believe that it was somehow missing from the older version of CASA I had been using. Apparently, it is not possible to use the "gridmode='aprojection'" parameter with the 'im.clean' task, which was the source of my confusion previously. Instead, gridmode='aprojection' must be used with the 'clean' task alone. The attached CASA python script 'CASA_imager_SKA_2src_aproj.py' shows my attempts at experimenting with the

gridmode='aprojection' algorithm option within the clean task (with earlier attempte commented out), i.e., I tried to form an image with CASA using/implementing the CASA 'aprojection' algorithm using a measurement set that had visibilities corrupted by the cortes beam model (with Meqtrees). However,in every case (i.e., different combinations of other parameters called within the 'clean' task), I would receive error messages such as the following when I tried to implement 'aprojection',

"Exception Reported: std::bad_alloc"

or

"SEVERE Exception Reported: PSFZero SkyEquation:SF calculation results in a PSF with its peak being 0 or less!"

Perhaps these error messages are due to the fact that the CASA 'aprojection' algorithm in its current form is specifically geared towards correcting for (E)VLA polarization squint and enabling gridding-based correction for RL beam-squint in off-axis systems such as the VLA, and so incompatable with or unable to correct for the Cortes beam pattern model. Or, perhaps I have somehow implemented gridmode='aprojection' incorrectly within the clean task CASA command.

3. Given that I haven't been successful so far in forming an image using the CASA 'aprojection' algorithm with a measurement set that includes simulated observed SKA visibilities corrupted by the cortes beam model, and/or it may be impossible to use the 'aprojection' algorithm in its present form to reverse the effects of the cortes beam model, the best way to go now, as you suggest, might be to install a new CentOS 5.x VM on myska and install the developer version of CASA on it. I thought it might be best to meet with you on Monday before getting started on that, but I can get started before we meet as necessary.