8-April-2013

Several changes to the HSI_CLEAN object have just been uploaded. There is no change to the default processing method of this algorithm. There have been improvements to some of the non-default methods used for determining the final images produced by Clean. We will soon begin processing tens of thousands of images (maybe many, many times more) using Clean so we felt it was worth the investment in time to make these changes. Even these modest changes required a number of hours in analysis, coding and testing.

There has been a change to the post-processing done by the Regress_Combine method. This method is done on the Clean_Component_Map and the Clean_Resid_Map to find an optimum method of combining the two. If CLEAN_REGRESS_COMBINE is set to 1, then the final map is found by doing a linear regression of the counts expected from the Component_Map against the observed counts. That finds a coefficient we'll call A. Next, the counts of the Resid_Map are regressed against the residual counts, the observed counts less those from the newly scaled (by A) Component_Map. (N.B. There is a profile method that we use to compute the counts expected from any map.) That gives a coefficient B for the Resid_Map. The final map will then be A*Component_Map + B*Resid_Map. This gives a much more robust, and we believe better, map than the previous version of Regress_Combine (regression on Component_Map and Resid_Map counts to find A and B in one operation). It's also better than the default map of Component_Map + Resid_Map because the scaling of the Component_Map suffers when the intrinsic source size is larger than the grid FWHM. Leaving the control parameters at their default values returns the default map. To get the values of the previous version of Regress_Combine, you must set CLEAN_OLD_REGRESS_METHOD and CLEAN_REGRESS_COMBINE.

There has also been an improvement to the ability to control what happens during the post-processing of Clean in the case of making a single Clean image and seeing how it looks. You may change the CLEAN_BEAM_WIDTH_FACTOR and only the convolution of the sources with the beam will change as part of the post-processing. It won't go through the full Clean (in hsi_map_clean()) operation as the beam convolution only affects the appearance of the default map. However, if CLEAN_REGRESS_COMBINE is selected it will use the newly convolved map as the Component_Map in the operation described above. This allows a user to explore the outcome of using different values of CLEAN_BEAM_WIDTH_FACTOR. The profile display tools use the final map produced by Clean for their input. The CLEAN_REGRESS_COMBINE operations are all in post-processing and they also won't trigger entry into hsi_map_clean() which was our original and is our default method that gives us the output map.

Three other control parameters are also trapped when they are Set to inhibit any reprocessing. Normally, changing an object's control parameters causes the object to reprocess when it's getdata method is used. However, for CLEAN_SHOW_N_MAPS, CLEAN_SHOW_N_CHI and CLEAN_PROGRESS_BAR changing their value will only have an effect on how the results are displayed and the PROGRESS_BAR widget.

Finally, there is a new info parameter, CLEAN_INFO_CONTROL. This holds the values of the Clean control parameters at the time the last image was made. The state of that is examined to see if any of the changes in those control parameters requires a re-entry into the hsi_map_clean routine.

Richard and Kim