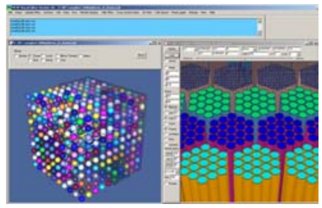

Graphical User Interface for High Energy Multi-Particle Transport

Quarterly Tech Report Number 3
(Modified for internet availability)
June 1st 2007 – August 31st 2007

PREPARED BY:
Visual Editor Consultants
P.O. Box 1308
Richland, WA 99352-1308
PHONE: (509) 539-8621
FAX: (509) 946-2001
Email: randyschwarz@mcnpvised.com
Home Page: http://www.mcnpvised.com

Contract Number: NNL07AA10C
August 2007

Project Summary

Computer codes such as MCNPX now have the capability to transport most high-energy particle types (34 particle types now supported in MCNPX) with energies extending into the teravolt energy range. The efficient use of these types of Monte Carlo tools is very important for modeling the effects of space radiation on humans, spacecraft, and equipment. This work will generate a graphical user interface for high-energy multi-particle transport. With this innovation, users of the MCNPX code will have access to a powerful graphical user interface for efficient creation and interrogation of their input files, which will significantly reduce the amount of time required to create and debug input files.

Table of Contents

PAGE

List of Figures

PAGE

1.0 MCNPX Visual Editor Support for Multiple Particles

CMCAI has modified the Visual Editor Fortran patch file to provide additional particle information to the C++ code to allow the user to differentiate different particle types. Figure 1 shows the upgraded particle track-plotting panel. This panel can be accessed by selecting Particle Display->Plot Particle Tracks from the main menu.

Particle track plotting in the Visual Editor works by having the Fortran MCNPX code calculate the particle position and then projecting this position on the 2D plot window. Because the particles are projected on a 2D plot window plane, a sphere and a cylinder may look the same. The user may have to display several 2D slices to fully visualize the source region. The user can specify the distance from the 2D slice displayed in the plot window on which particles will be projected by changing the value of the "Distance from the plot plane (cm)" value. The default distance is 100 cm on either side of the 2D plot window plane.

In Figure 1, test input file inp110 has been read in. It has ten particle types identified in the mode card (mode n p | h / z d t s a). Within the "Particle Track Plotting" window, each particle type is listed. The user has a number of options that can be set for each particle type. The "show" column is used to request that the particle type be displayed when the collisions are plotted. This can be toggled on (indicated with an "X") and off with a left click of the mouse button. The default is to show all particles. The next column has a heading of "use" that indicates the min and max values, specified in the following two columns, are to be used to limit the particles displayed to the range specified by the user. This can be toggled on (indicated with an "X") and off with a left click of the mouse button. The default is to not use the specified ranges.

The min and max columns allow the user to specify an upper and lower value for the particles plotted. The meaning of these values depends on the "color by" option. The values are set by clicking on the "Min" or "Max" value for the particle type listed in the row. If "energy" is selected, the values represent a lower and upper energy, if "weight" is selected the values represent a lower and upper weight. The default is to use energy.

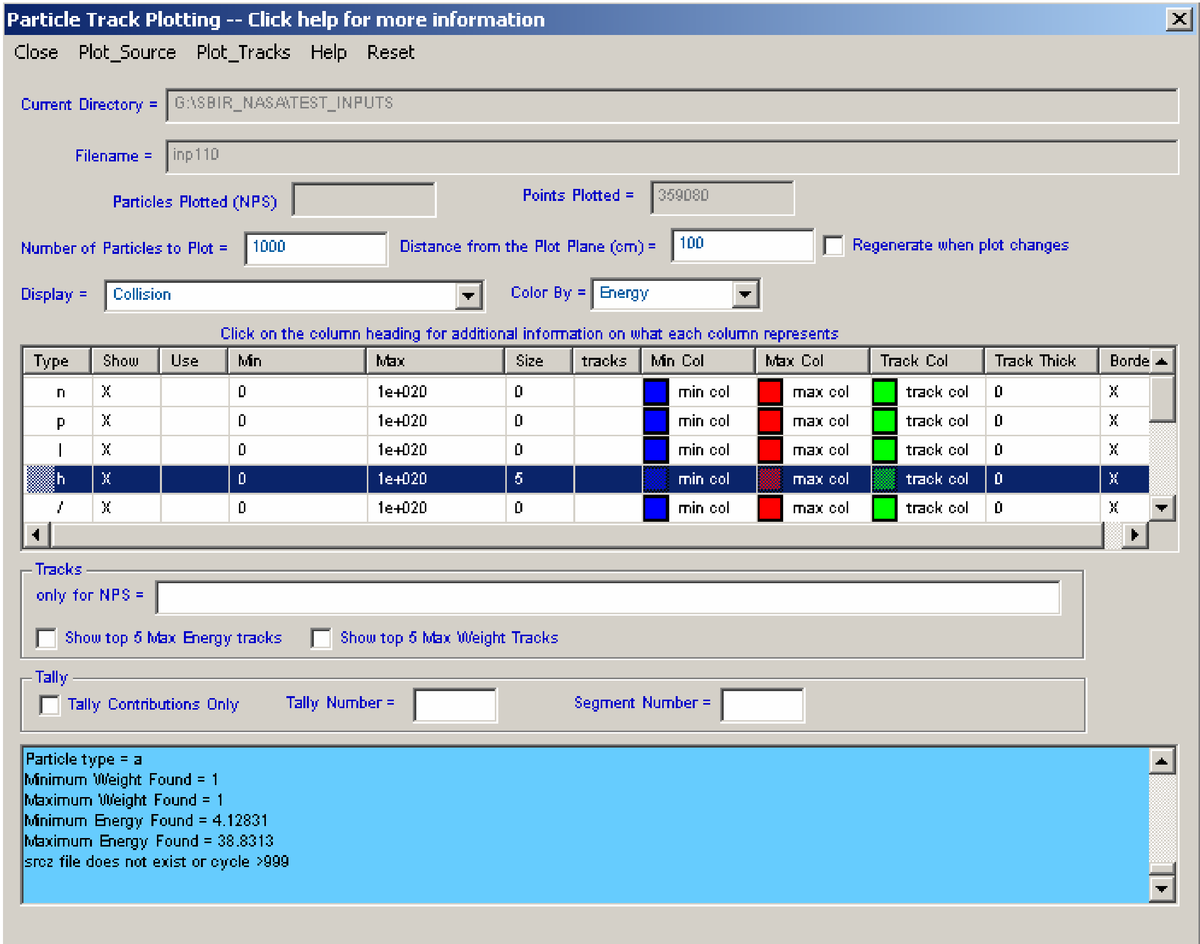

Figure 1. Particle Track Plotting Panel in the MCNPX Visual Editor.

The next column, labeled “size”, allows the user to set the size of the particle. The user can select a row in this column with the left mouse button and a menu will appear that will allow the user to change the relative size from any value between 0 and 5. The default value is 0, which will plot each point as a pixel.

Following the “size” column is a “tracks” column. If this column is selected, the user is requesting that tracks be plotted for the particle selected. This can be toggled on (indicated with an “X”) and off with a left click of the mouse button. The default is to not plot the particle tracks.

The next three columns allow the user to set the minimum color, the maximum color, and the track color. When the user selects a row in this column, a panel will be displayed that allows the user to specify the color for each of these three entries.

The “Track Thick” column allows the user to specify the thickness of the track line if the user has requested that particle tracks be plotted. The user can select a row in this column with the left mouse button and a menu will appear that will allow the user to change the relative size from any value between 0 and 5. The default value is 0, which will plot the thinnest line for the particle tracks.

The final column allows the user to set a black border around the collision points. The default is to show this border for all particle sizes greater than 0.

In Figure 2, the source for inp110 is plotted by selecting "Plot_Source" from the menu at the top of the particle track-plotting panel previously shown in Figure 1. The source definition is defined as follows:

```
sdef sur = 1 erg = 800 par = h dir = 1 pos = 0 0 0 rad = d1
si1  2
```

This defines a source on surface 1 at x=0, y=0, and z=0. It has a radius of 2 cm defined by "si1". The particles are generated in the positive z direction, because this is a surface source and the default reference vector (VEC) is in the positive direction of the surface normal, which is (0,0,1) for surface 1. The dir vector indicates that all particles are generated in this normal direction.

The source particle is a proton (h); to see the source it is necessary to set the particle size to 5 for protons and then select Plot_Source.

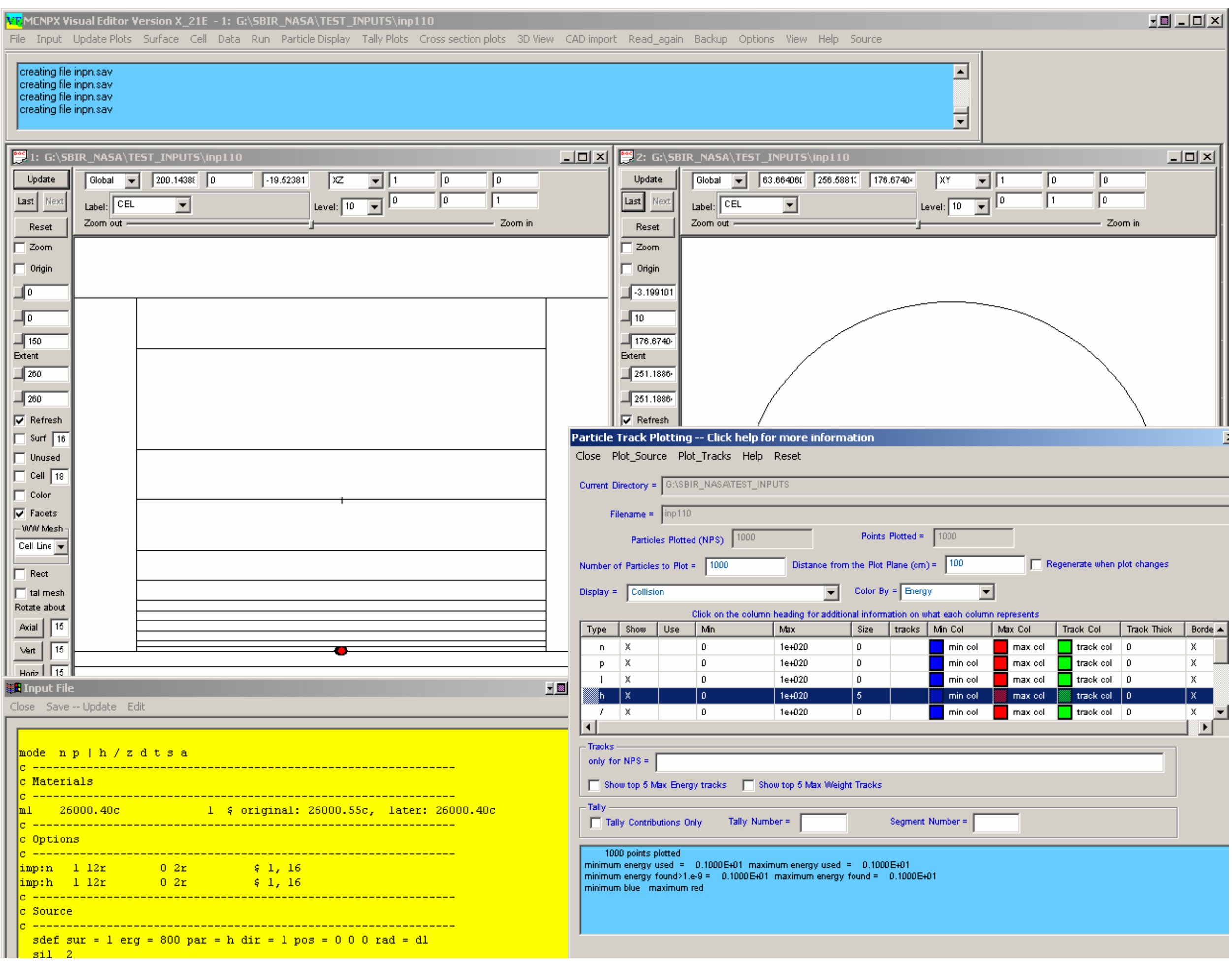

Figure 2. Plotting the Source in the MCNPX Visual Editor.

Figure 3 shows a close-up view of the surface source. When the particles are plotted they now appear as a line instead of a point. The line represents the surface source viewed from the side.

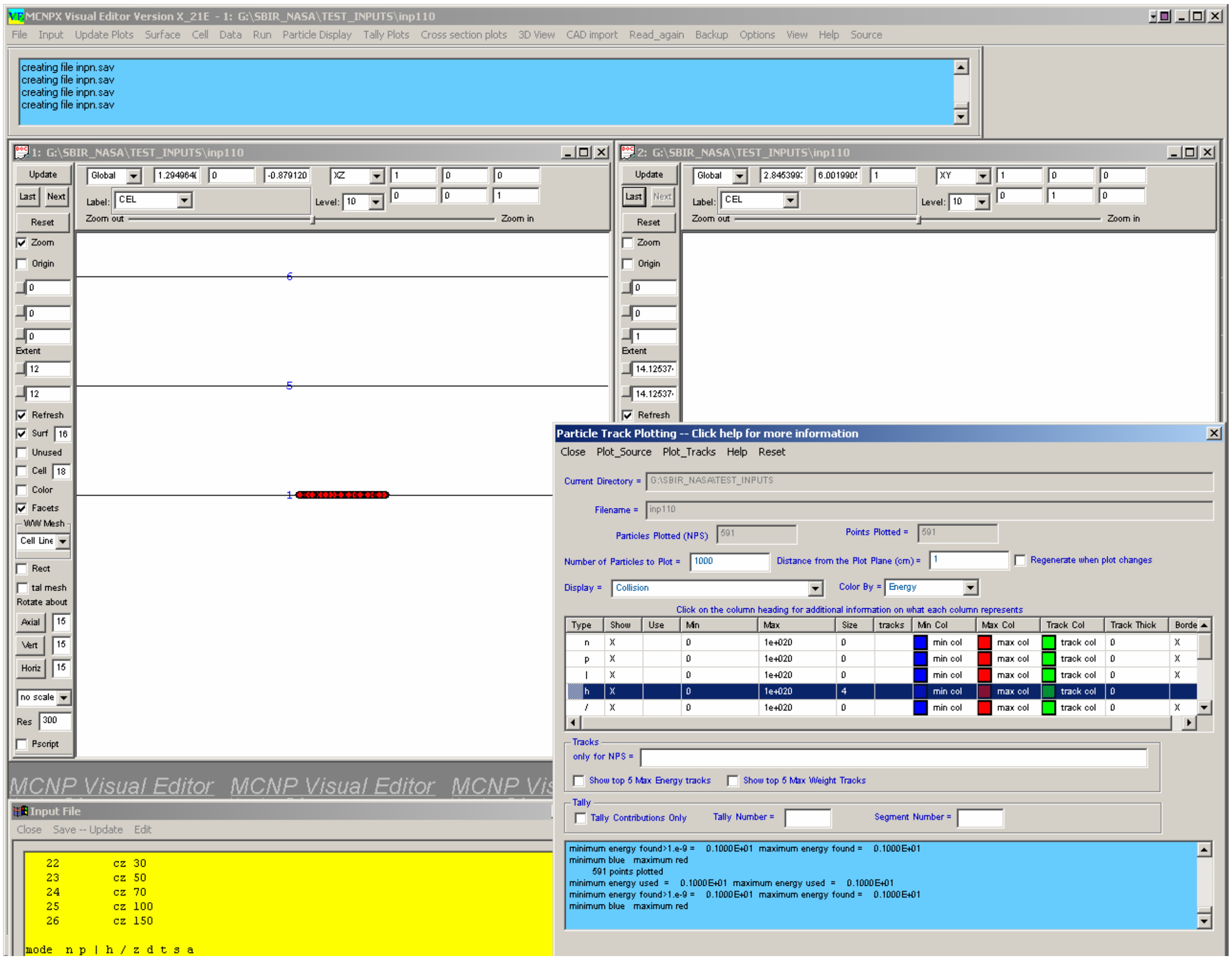

Figure 3. Plotting the Source in the MCNPX Visual Editor with Equal Extents.

In Figure 4, "Plot tracks" is selected to plot the particle tracks. Because the source particle (protons) is enlarged (set to size 5), the protons can be observed to be moving upward and generating the secondary particles that are still set to a pixel size. The low energy is set to blue, and the high energy is set to red.

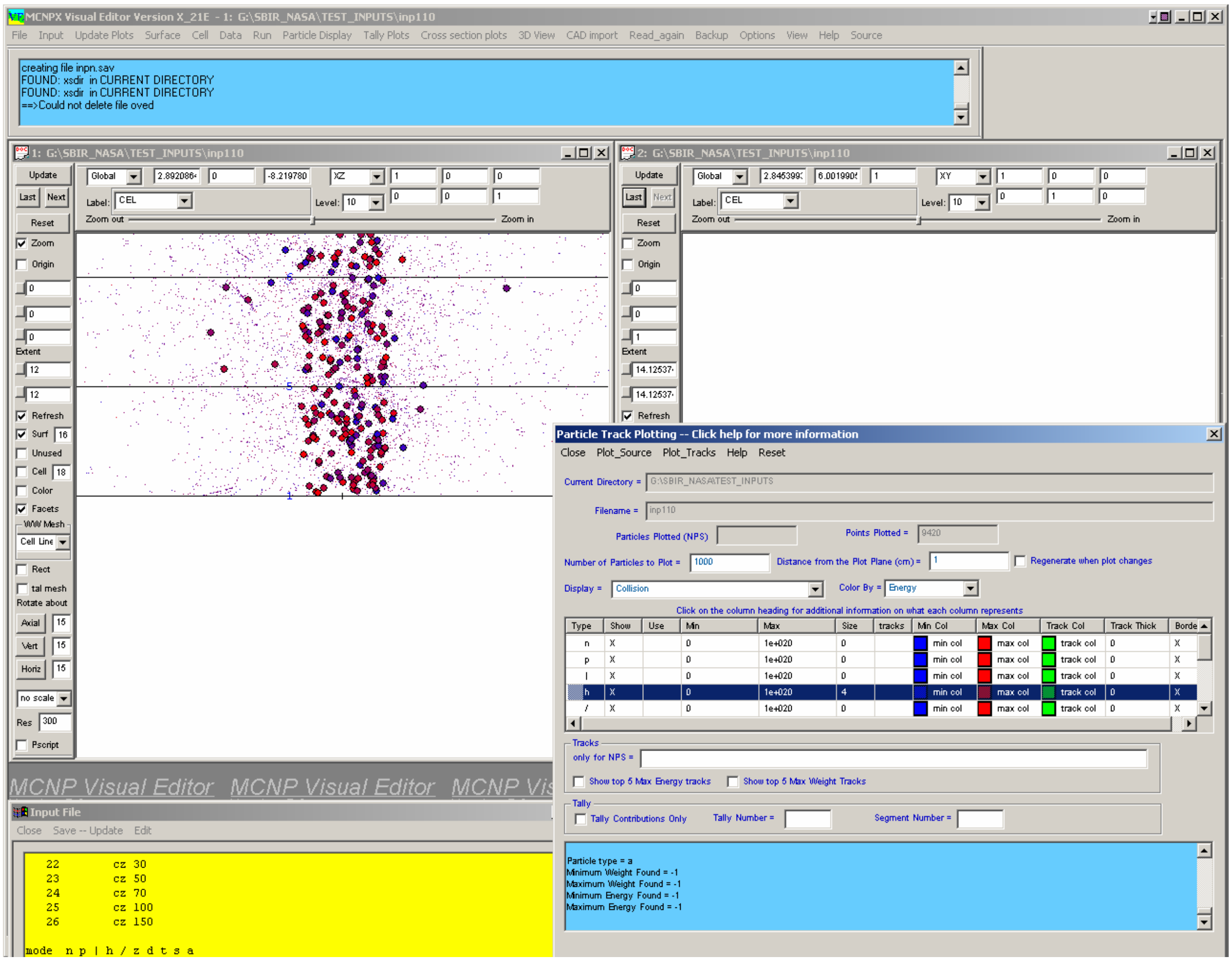

Figure 4. Plotting Collision Points.

In Figure 5 the particles are colored by weight and the extents are set to 100. To differentiate the different particles that are generated, the neutrons are yellow, the photons are green, and the muons are cyan colored. The color of the particle is set by clicking on the color box and setting it to the desired color. The color picker is shown in Figure 5.

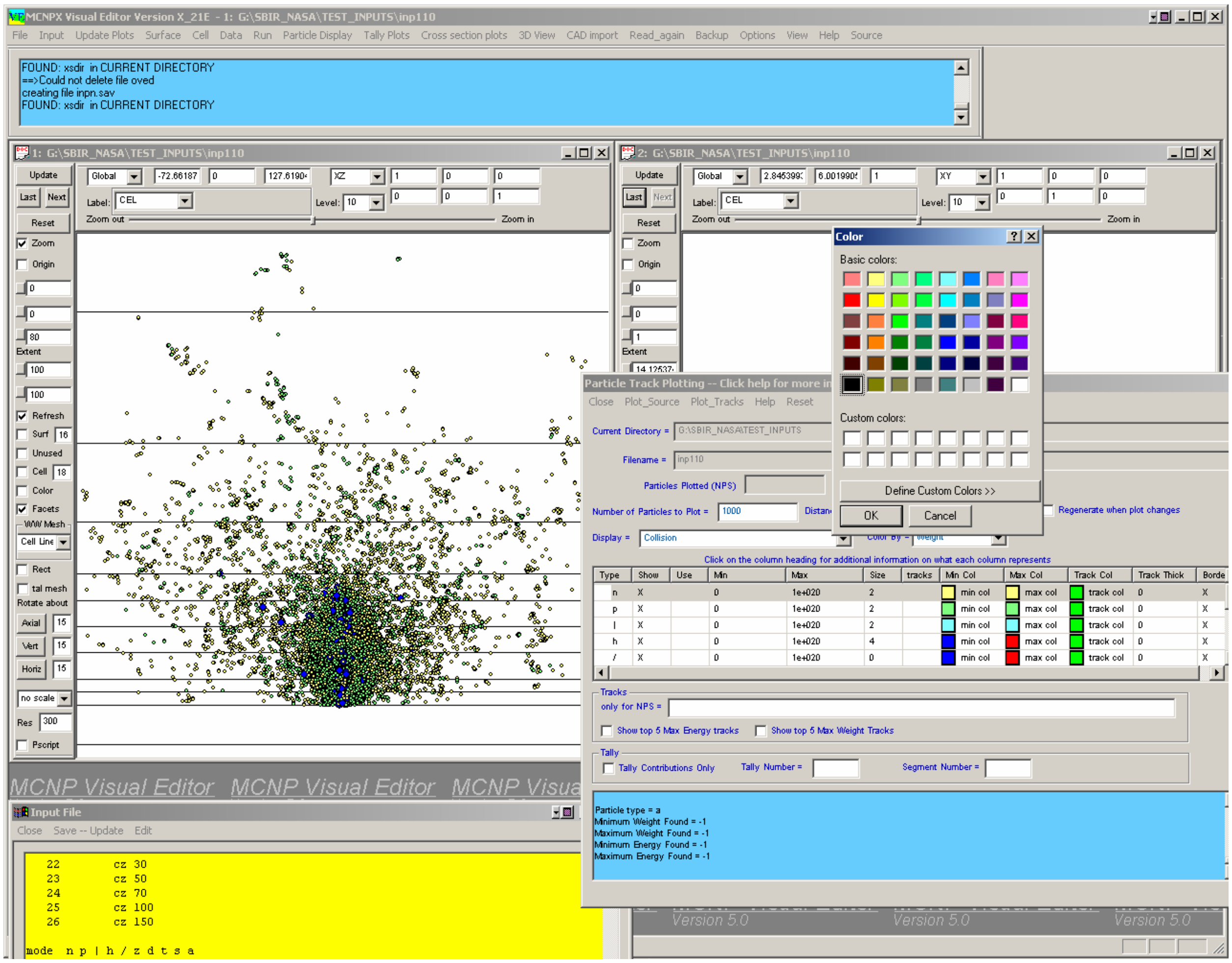

Figure 5. Plotting Collision Points with Different Colors.

In Figure 6, only the source protons and the generated neutrons are shown. All of the other particles are hidden (by unselecting the show option). The source protons are red and set to the smallest size, while the neutrons go from blue to green (depending on the energy) with a size of 3. The distance from the plot plane is set to 10 cm.

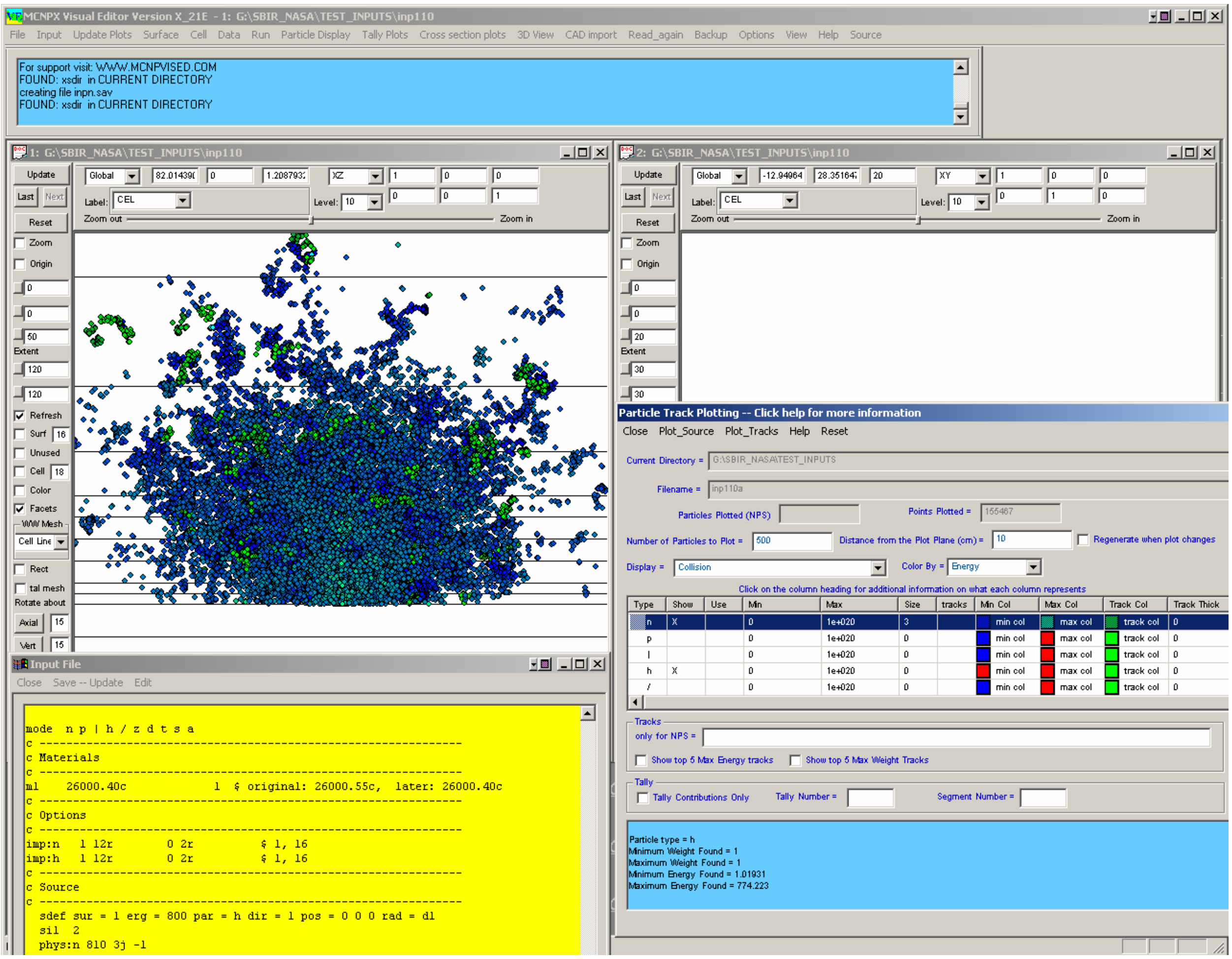

Figure 6. Plot of Neutron Collision Points.

Figure 7 shows the photons generated for this input file by selecting photons and increasing the size of the photon particles.

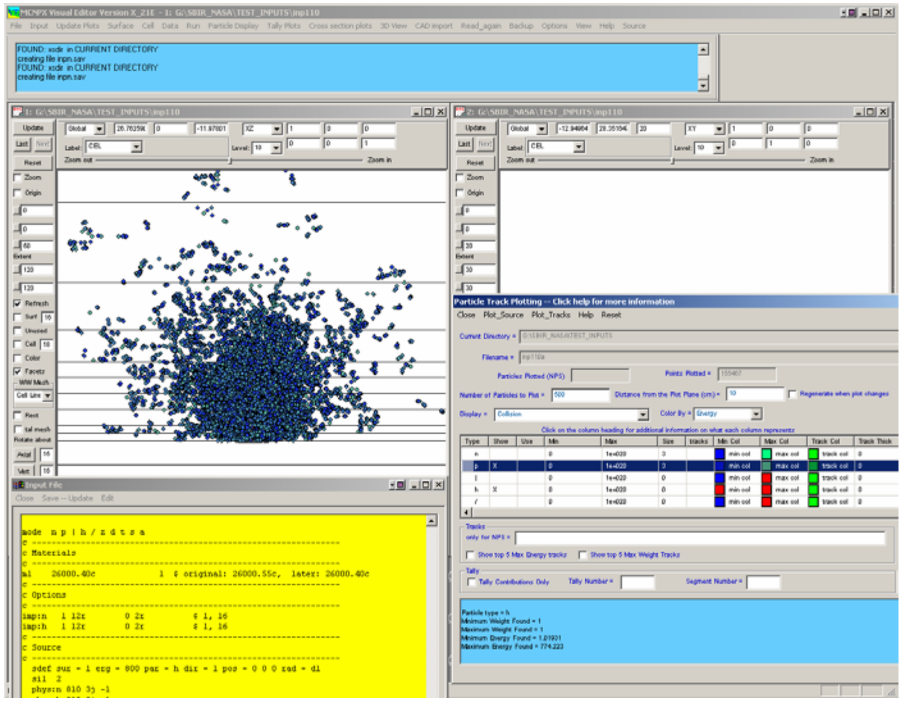

Figure 7. Plot of Photon Collision Points.

Figure 8 shows the muons generated for this input file.

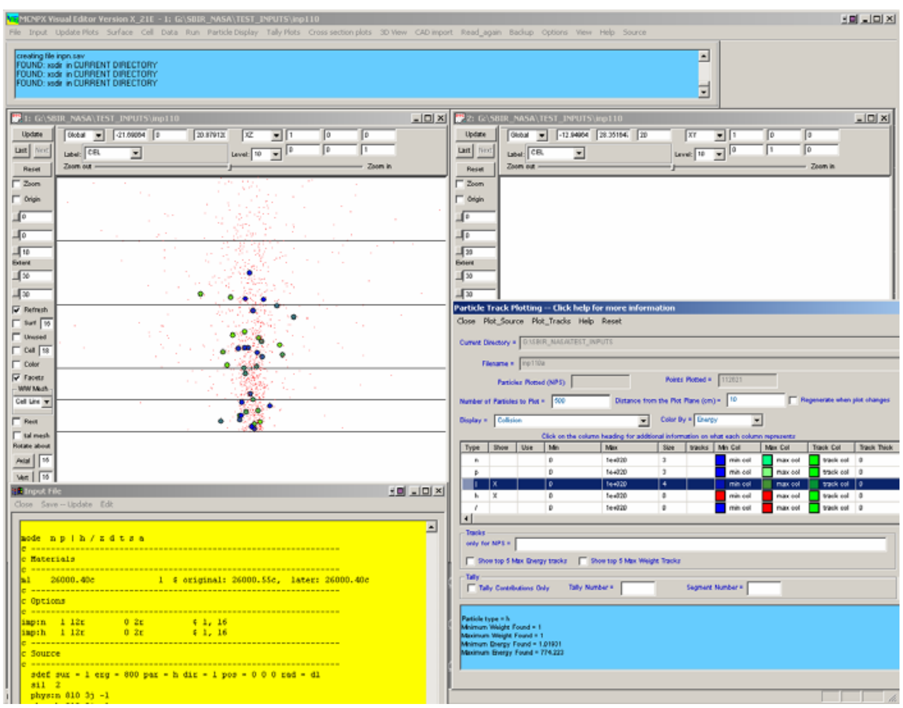

Figure 8. Plot of Muon Collision Points.

Figure 9 shows the +/- pions generated for this input file.

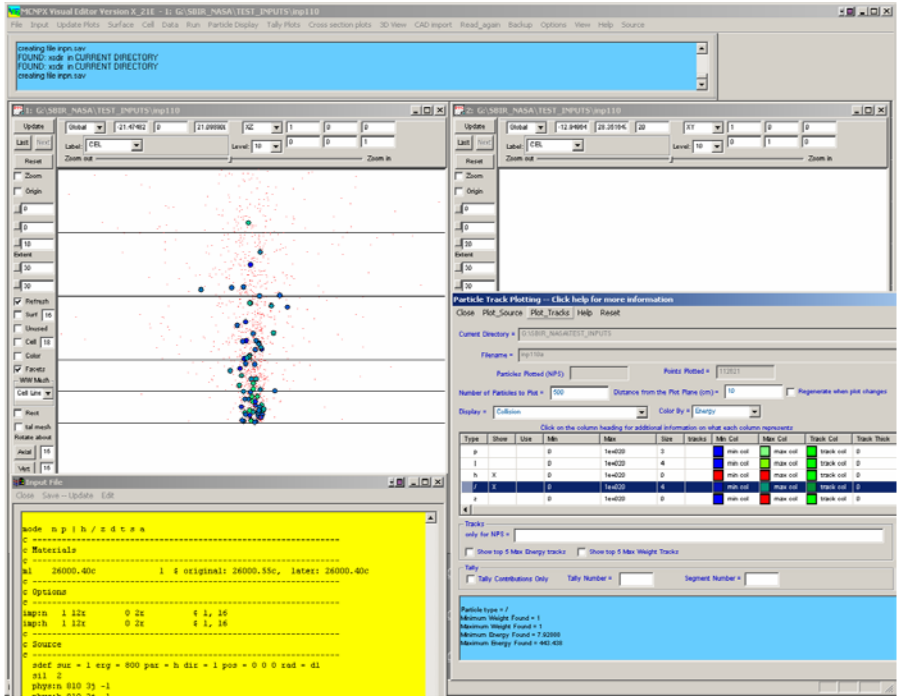

Figure 9. Plot of Pion Collision Points.

Figure 10 shows the neutral pions generated for this input file.

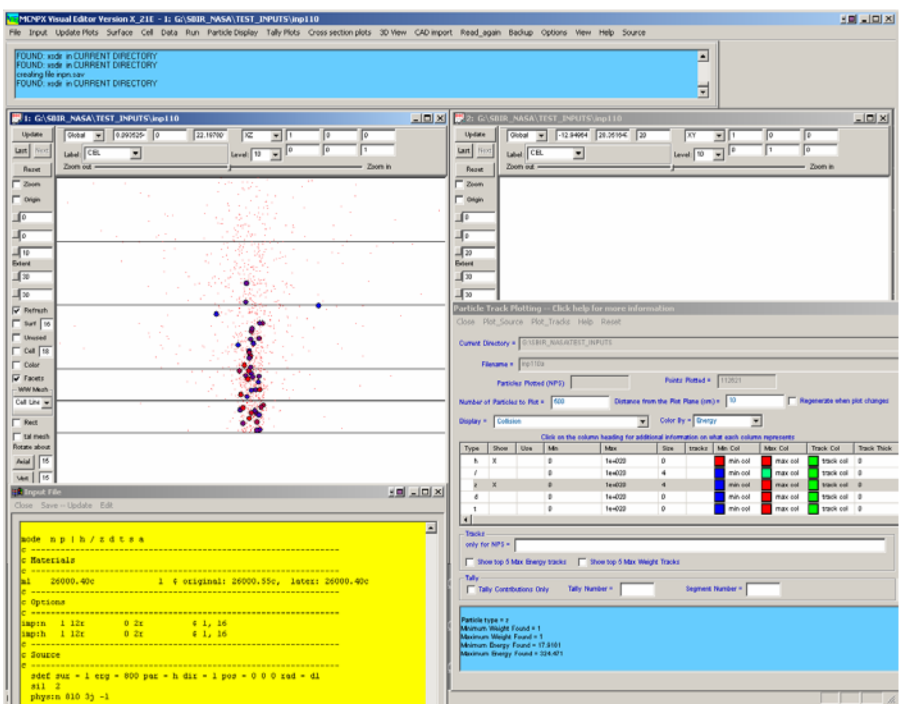

Figure 10. Plot of Neutral Pion Collision Points.

Figure 11 shows the deuterons generated for this input file.

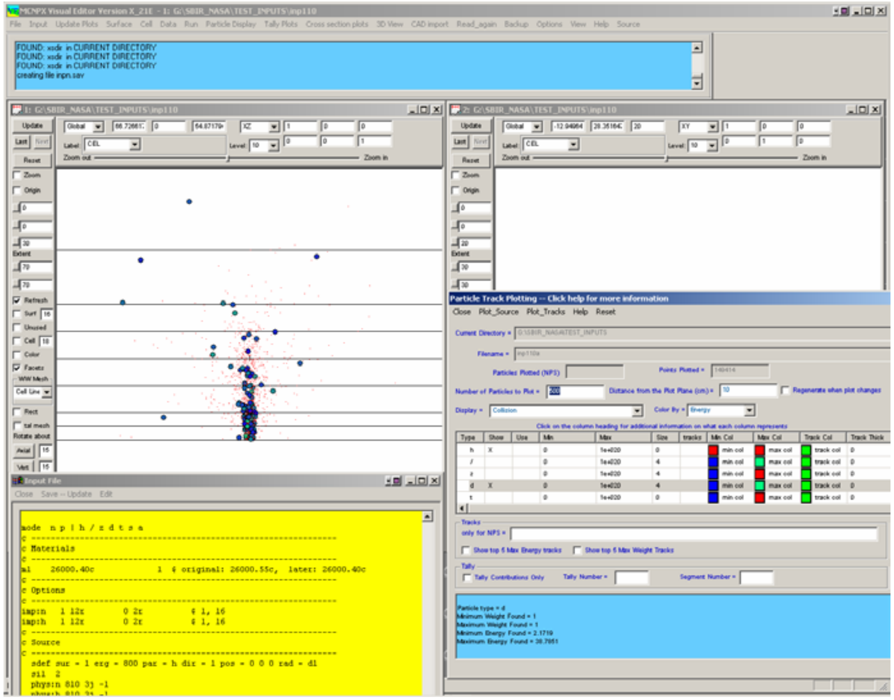

Figure 11. Plot of Deuteron Collision Points.

Figure 12 shows the tritons generated for this input file.

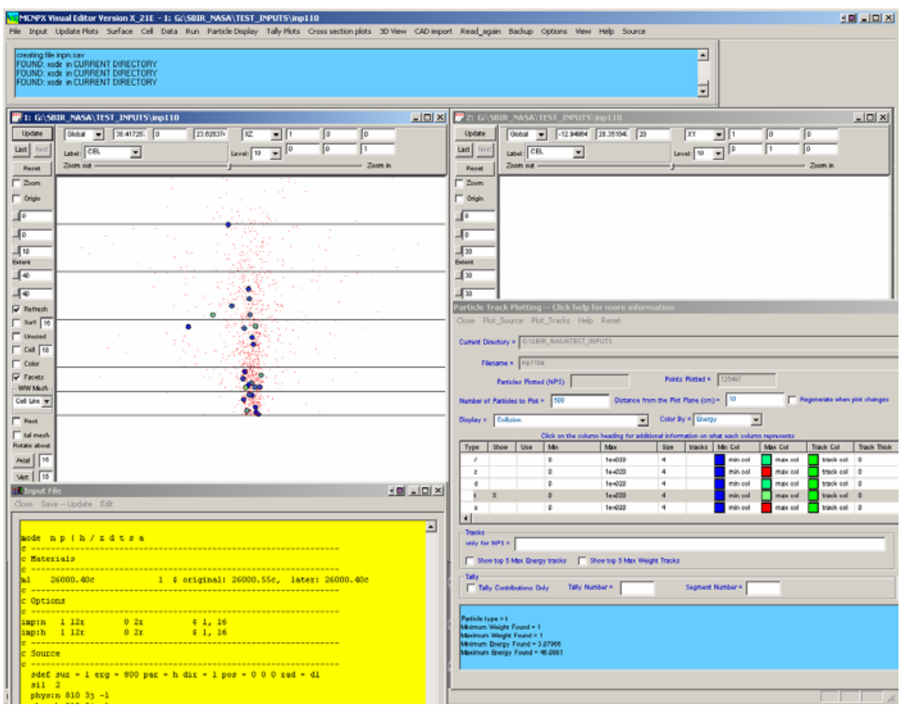

Figure 12. Plot of Triton Collision Points.

Figure 13 shows the He-3 collision points generated for this input file.

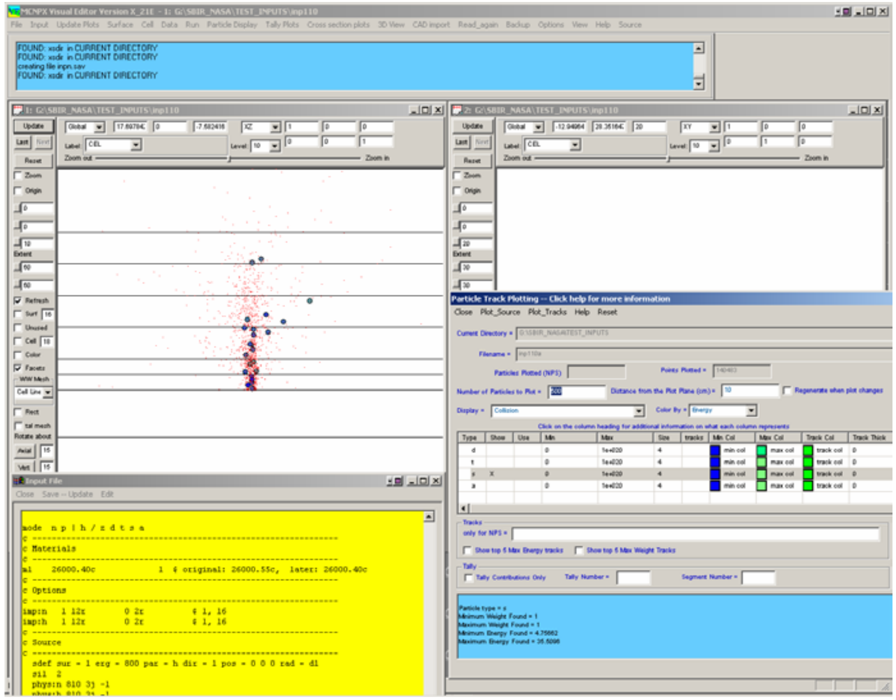

Figure 13. Plot of He-3 Collision Points.

Figure 14 shows the He-4 collision points generated for this input file.

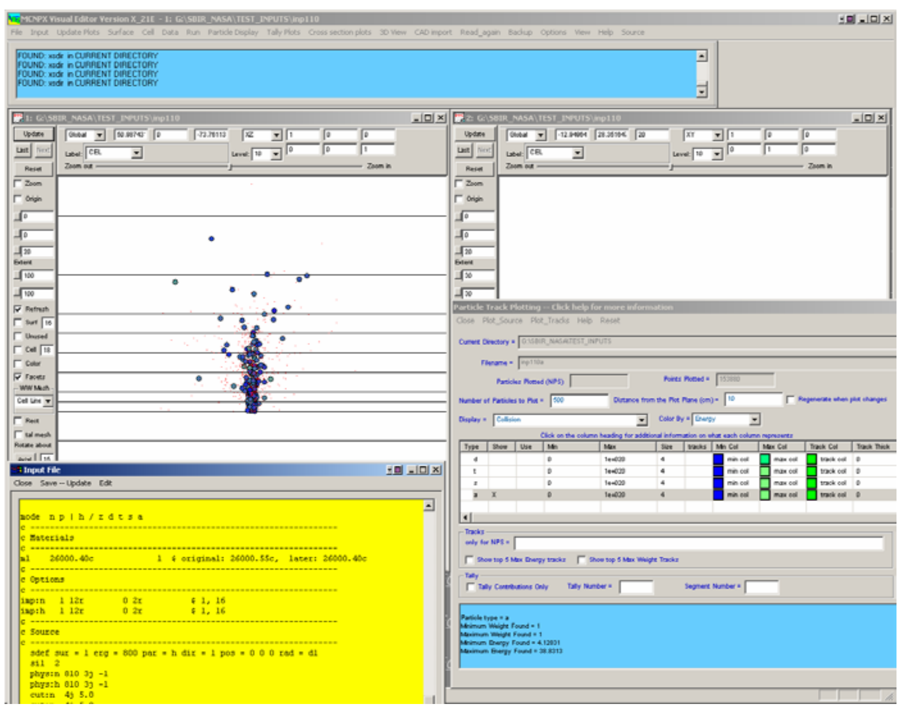

Figure 14. Plot of He-4 Collision Points.

Figure 15 shows the He-4 collision points generated for this input file. The collision points have been set to a cyan color, tracks have been turned on, and the color of the track line has been set to magenta.

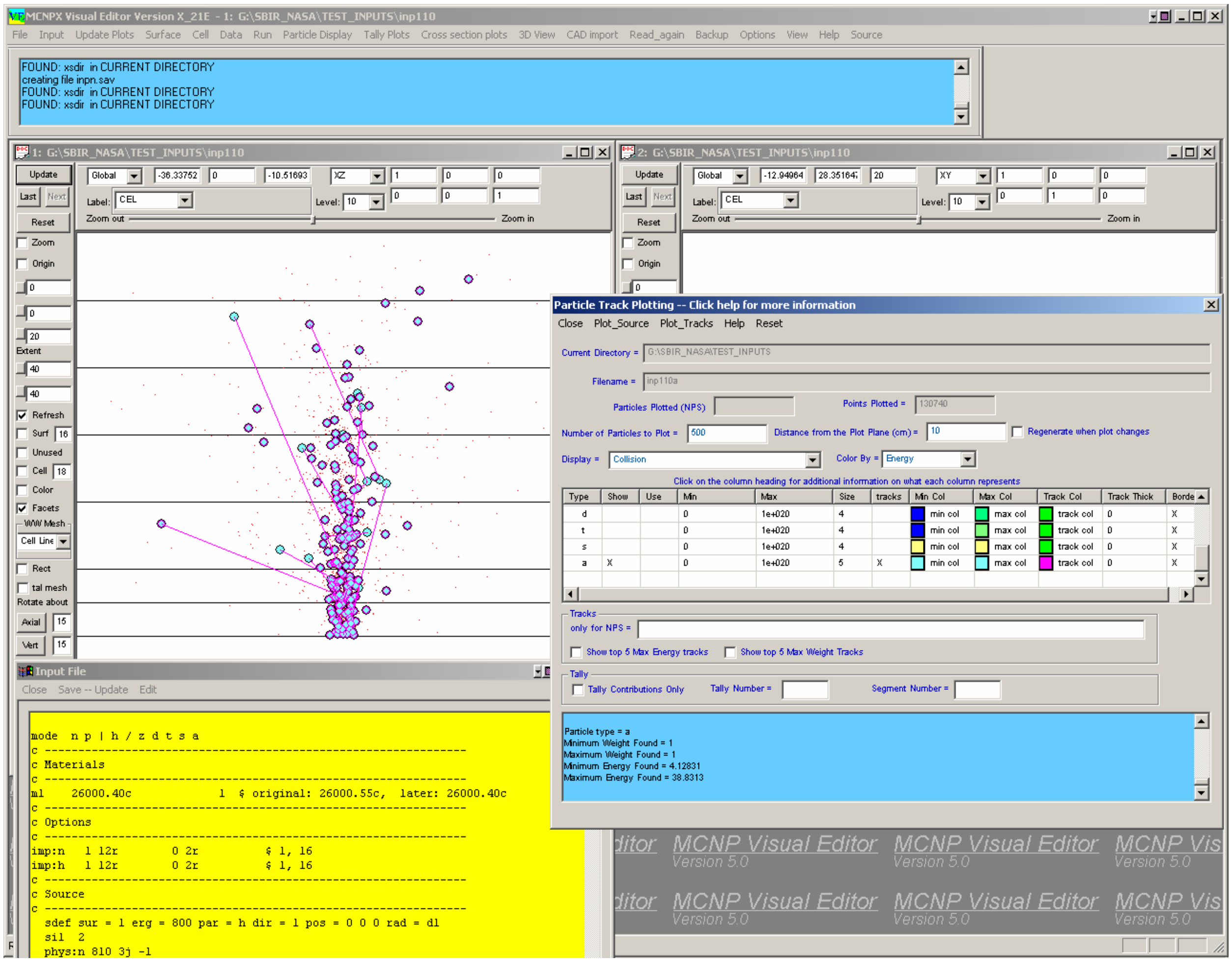

Figure 15. Plot of He-4 Collision Points Including Tracks.

Figure 16 shows photons in a cask with the particles colored by weight. The particles are biased through the cask shield and can be observed going from a high weight red color to a low weight blue color as the biasing in the shield lowers the weight.

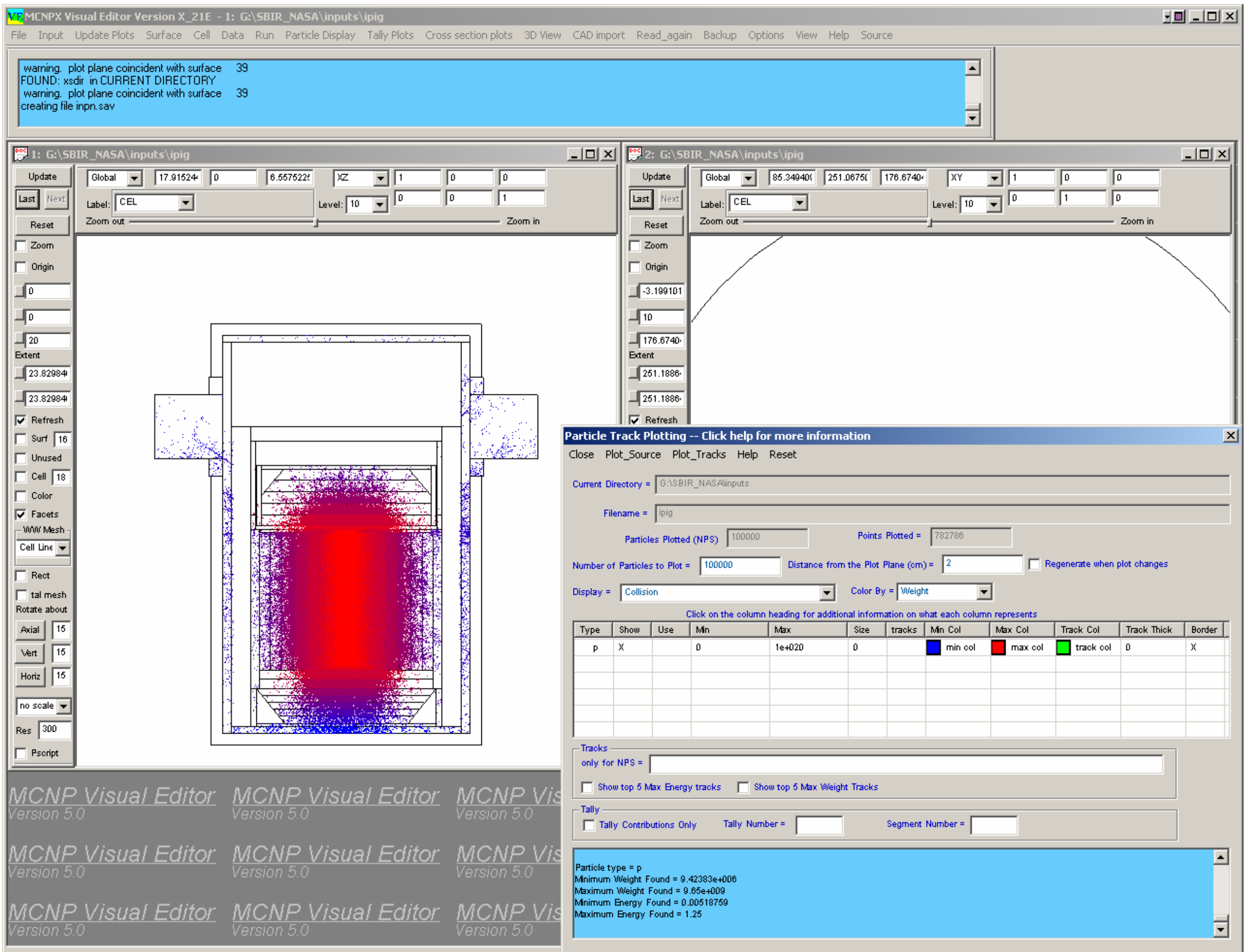

Figure 16. Particles Colored by Weight in a Cask.

In Figure 17, the “Tally Contributions Only” checkbox has been selected to only show those particle tracks that eventually lead to a contribution to a tally are shown. The tally in this case is on the outside of the cask.

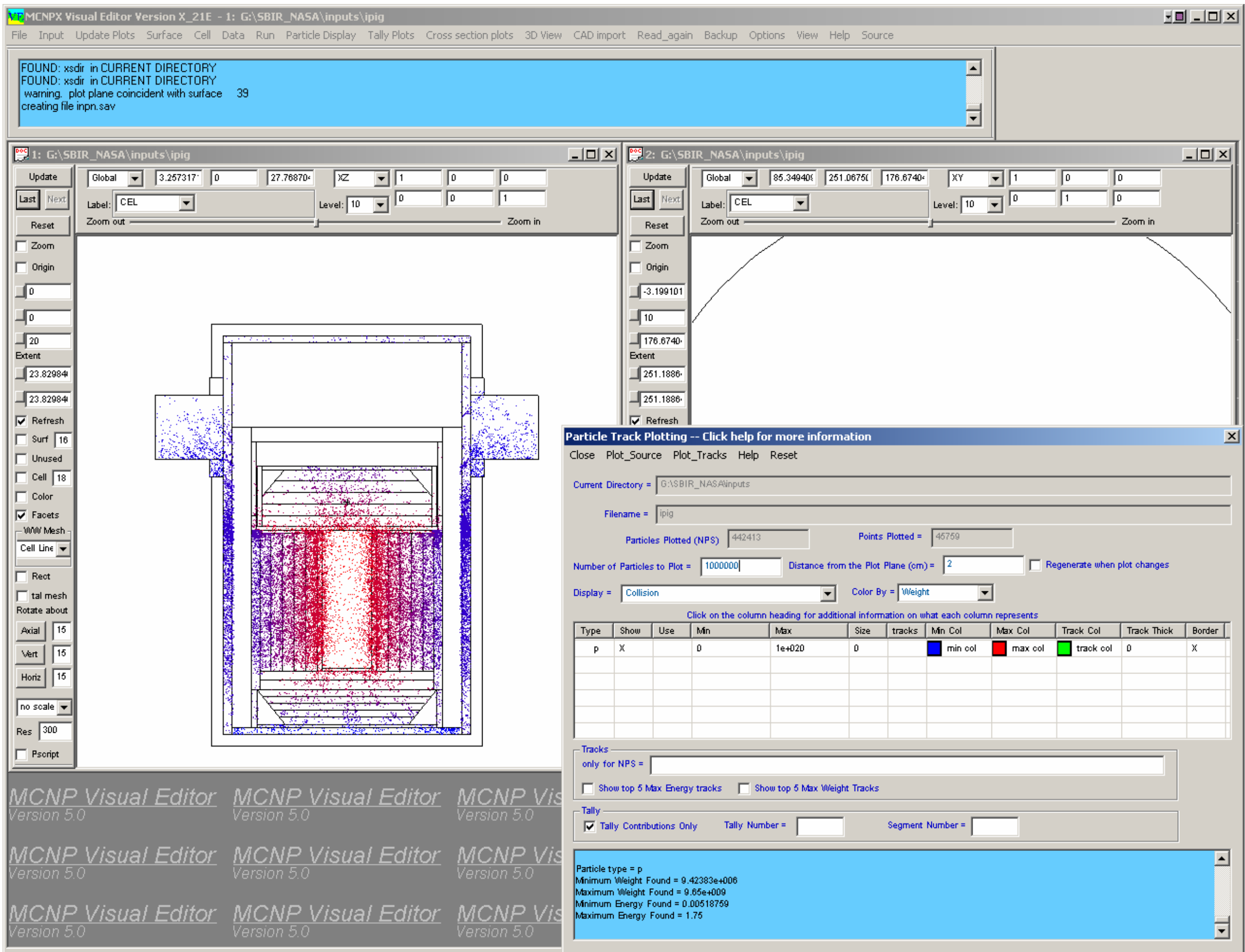

Figure 17. Particles that Contribute to a Tally Outside the Cask.

In Figure 18, the high weight particles have been selected by setting the range of weights to plot between 4e9 and 1e10. The weights are high because they have been multiplied by a weight scale factor in the source of 9.65e9. High weight particles can be observed traveling through a steel streaming path (shown in yellow) and going to the outside tally region of the cask. Because the steel is less effective as a shield than the lead, the particles can go through this steel region more easily than through the lead cask.

This additional information indicates that the dose rate near the top of the cask should be higher than the dose rate outside the center of the cask. Additional calculations show that for this case, the dose rate is a factor of two higher in this region. This is an example of how being able to display particle tracks can lead to additional insight into the problem being analyzed.

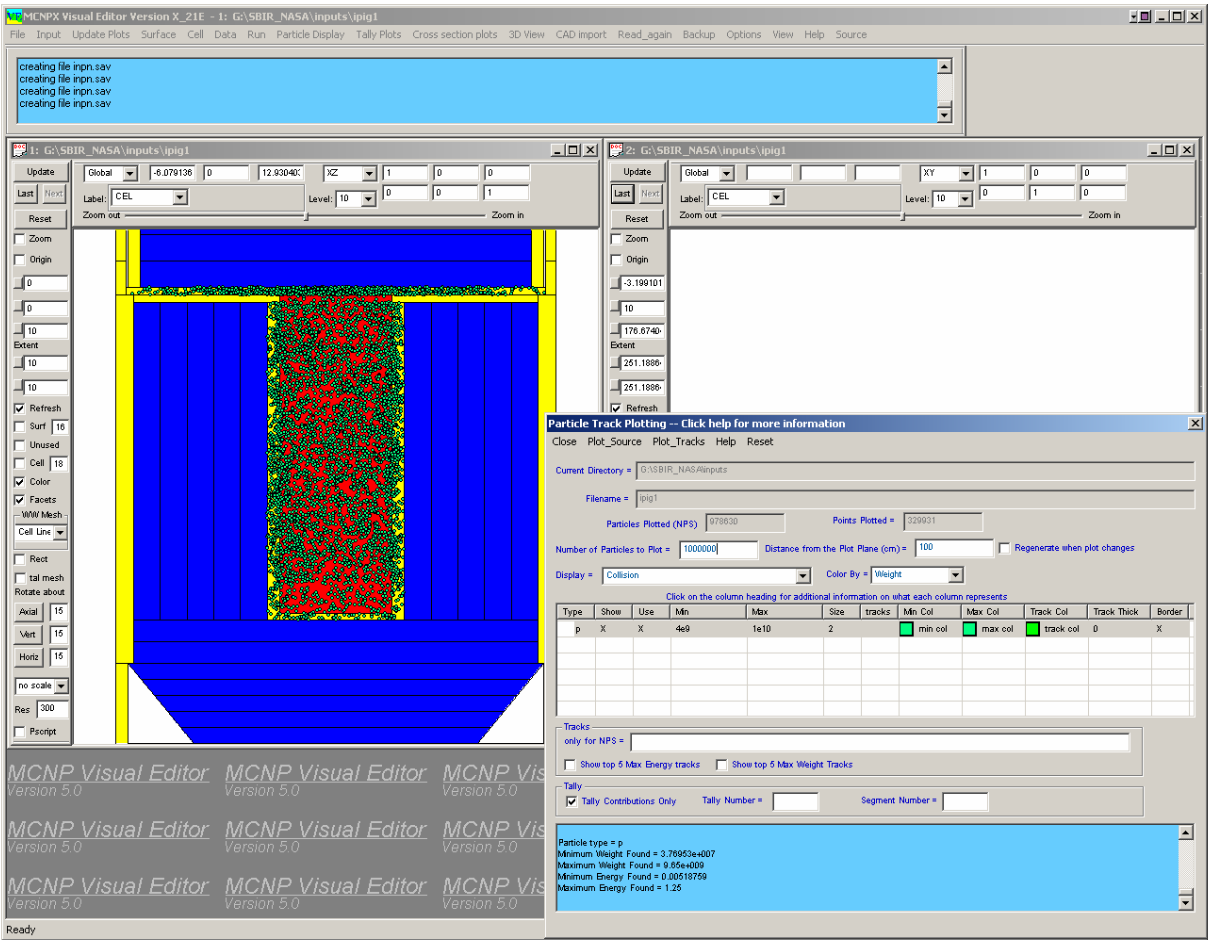

Figure 18. Plot of High Weight Particles Contributing to a Tally Region.

Figure 19 shows a cask containing cesium capsules with a cylindrical air chamber in the top left corner of the cask. This plot shows the source and collision points colored according to weight. The biasing of the source region can be observed. The left pin has more source than the right pin. The upper regions of both pins have more source points than the lower regions. The source biasing lowers the weight.

As the particles go through the steel shield, the particles are split using importance biasing that further reduces the weight. The end result is a plot showing the particles being biased towards the air chamber and changing from red to blue as they lose importance. This plot verifies that the particles are going in the expected direction.

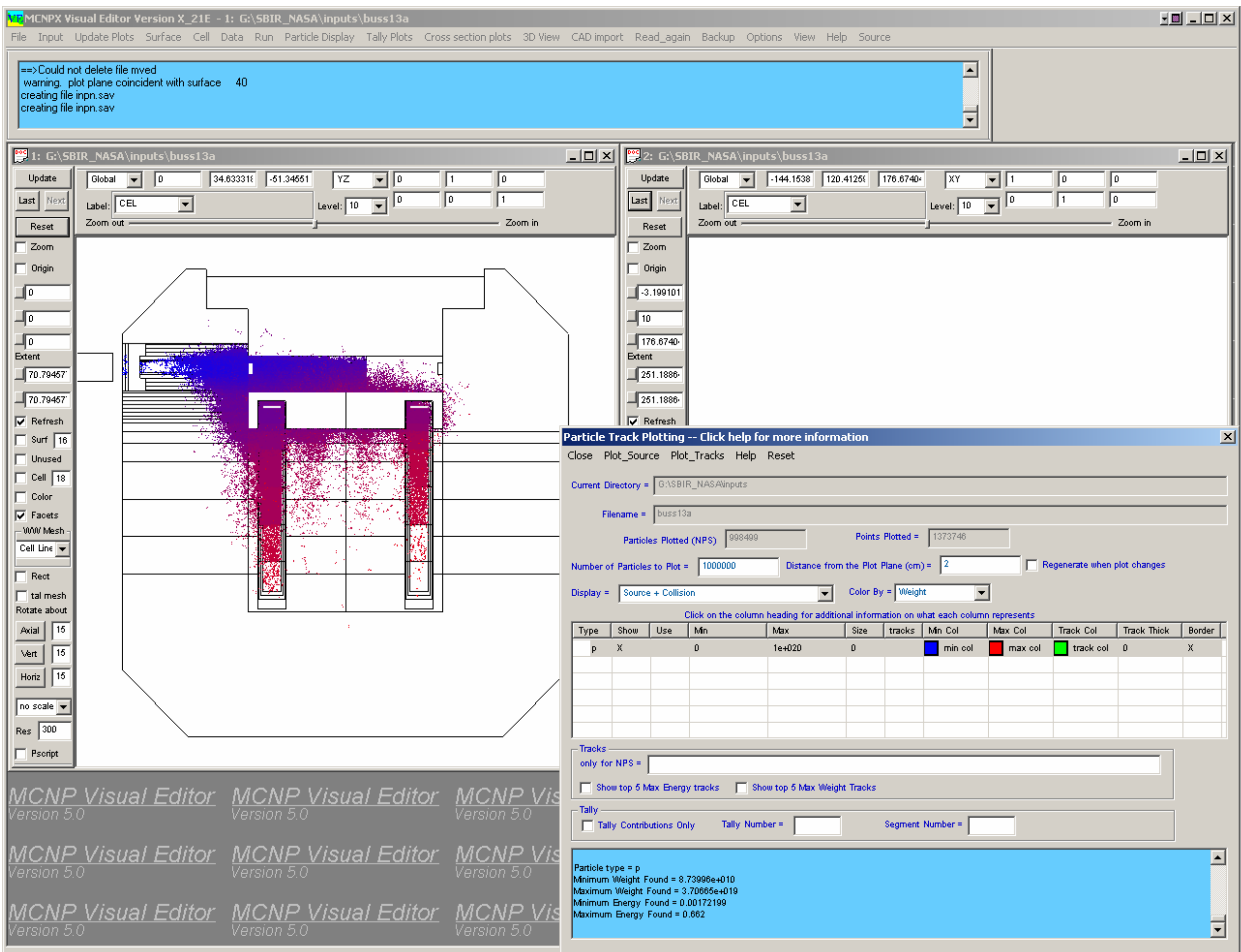

Figure 19. Plot of Biasing Particles Toward an Air Gap in a Cask.

Figure 20 shows the same cask but now only shows those particles that contribute to the detector outside the cask.

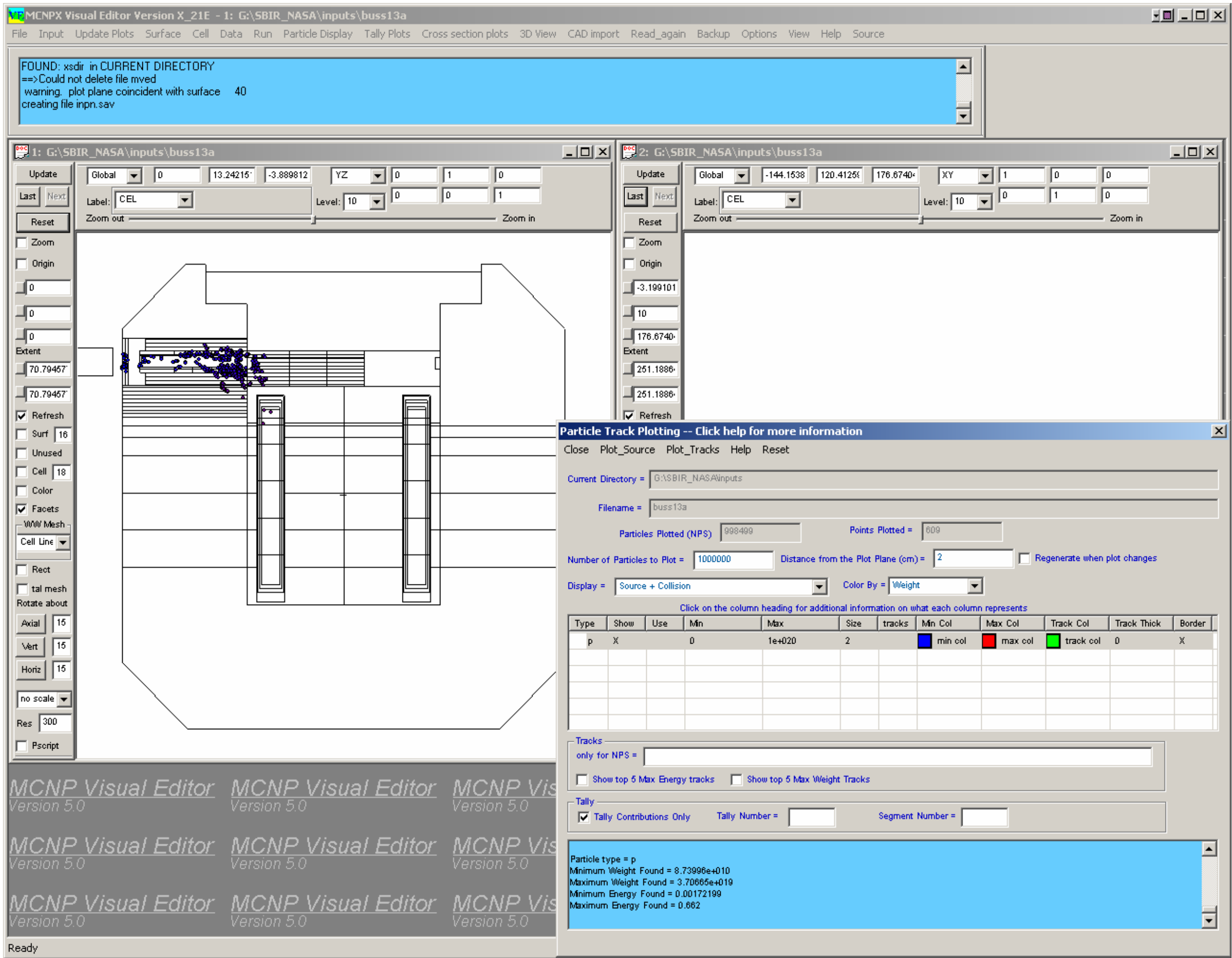

Figure 20. Plot of Particles that Contribute to the Tally Region.

2.0 MCNPX Visual Editor Mesh Tally Plotting

Tally plotting has been significantly updated to display mesh tallies. Tally mesh plotting has been partially implemented and the run-time tally has been updated. Additional work is required to make sure all of the tally plotting features work.

Figure 21 shows the revised tally-plotting panel for a 2D plot. The different tally-plotting modes are now contained in tabs to only show the information for the tally being plotted.

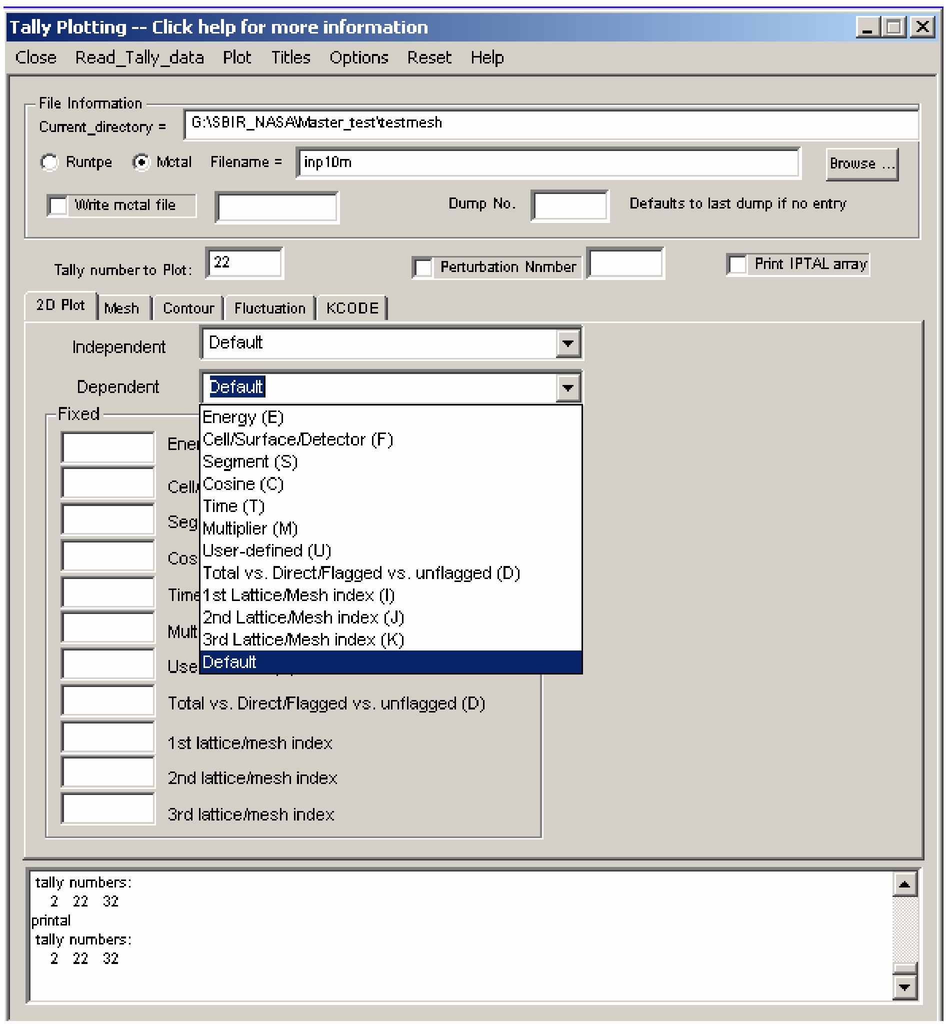

Figure 21. Updated Tally Plotting Panel.

Figure 22 shows the run-time plotting capability. The input file is read in and initialized in the run-time tally panel by selecting the input file and selecting "initiate". Once the file has been read in and has run for an initial number of particles (default 5,000), the user can pull up the tally plotting panel to set up the tally to plot. Once the tally has been set up, the user can go back to the run-time tally panel and either run for one additional cycle (default 5,000 particles) or run continuously (plot gets updated every 5,000 particles). The user can stop the running code by selecting "stop" from the run-time tally panel. Once the code has been stopped, the tally being plotted can be modified.

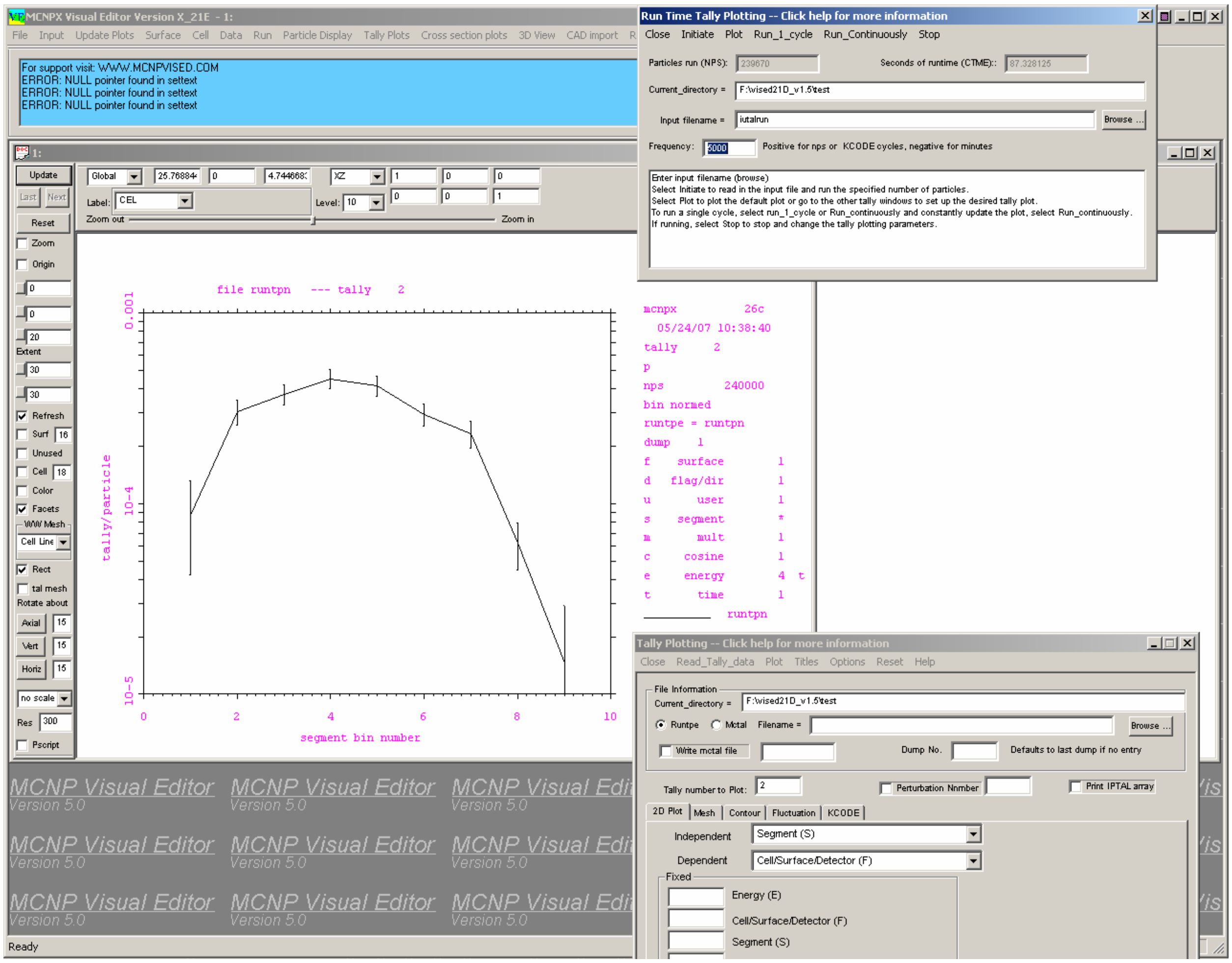

Figure 22. Run-Time Tally Plotting.

Figure 23 shows a mesh tally plot of a critical configuration of seven barrels of fissionable material. This input file is the same one described in the MCNPX release notes for version 2.5.f (LA-UR-05-0891). To generate this plot the code was run and a mctal file generated. The tally plotting window was then brought up by selecting Tally Plots->2D Tally Plots from the main menu.

At the command prompt this plot is generated using the commands “rmctal <mctal fname>” followed by “tal 12 free ik”. In the Visual Editor, the user must first read in the mctal file by entering the name of the mctal file, selecting the “mctal” option, and then selecting “Read_tally_data” from the main menu. Next the user must set the tally number to 12 and then select the mesh tab to bring up the mesh tally plotting options. In the mesh tally window, the user must select IK from the pull down menu to specify the view to plot. Finally, the user must select the “Plot” menu option to plot the mesh tally. The user can then set the window to rectangular by selecting the “Rect” option in the plot window.

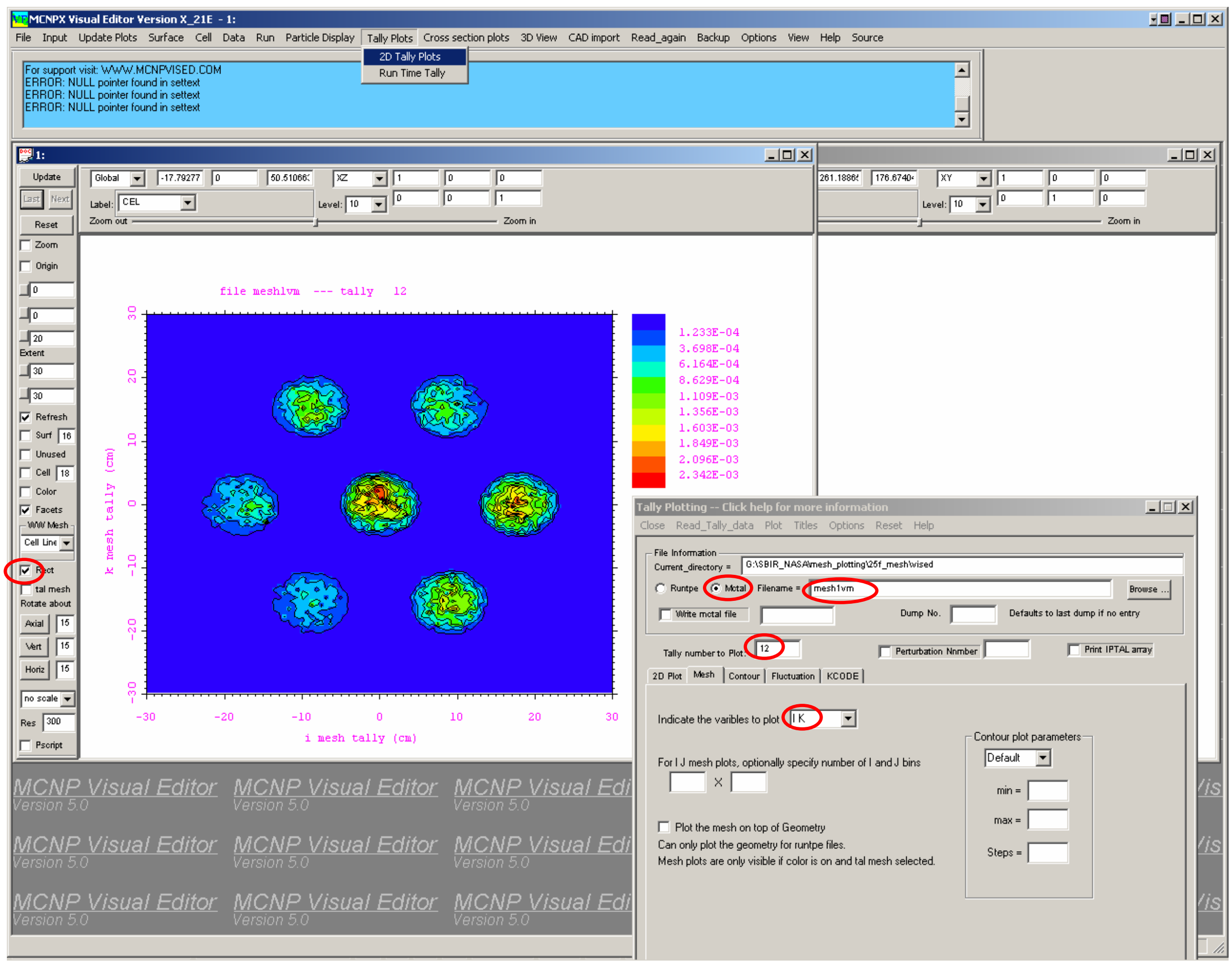

Figure 23. Mesh Tally Plot in the MCNPX Visual Editor.

Figure 24 shows a plot of the same mesh tally plotted on top of the geometry. This plot is also taken from the plot shown in MCNPX release notes for version 2.5.f (LA-UR-05-0891). To generate this plot from the command prompt the user first reads in the runtpe using "Runtpe = <runtpe filename>" followed by "PLOT" to get into the geometry plot mode. Then the user enters the command "py 4 ex 40 or 0 4 0 la 0 1 tal 12 color on la 0 0 con 0 100 %".

In the Visual Editor, the user must read in the runtpe by setting the runtpe file name and then selecting "Read_tally_data". Next, the user must set the tally number to "12" and then select the "Plot the mesh on top of the Geometry" check box. This will then set the "talmesh" toggle button in the plot window and will also set the "color" to "on" in the plot window, because tally mesh plots will not be visible if color is not on. Next it is necessary to set the plot view so that the mesh can be seen. For this example, the Y origin is set to "4" and the extents are set to "40".

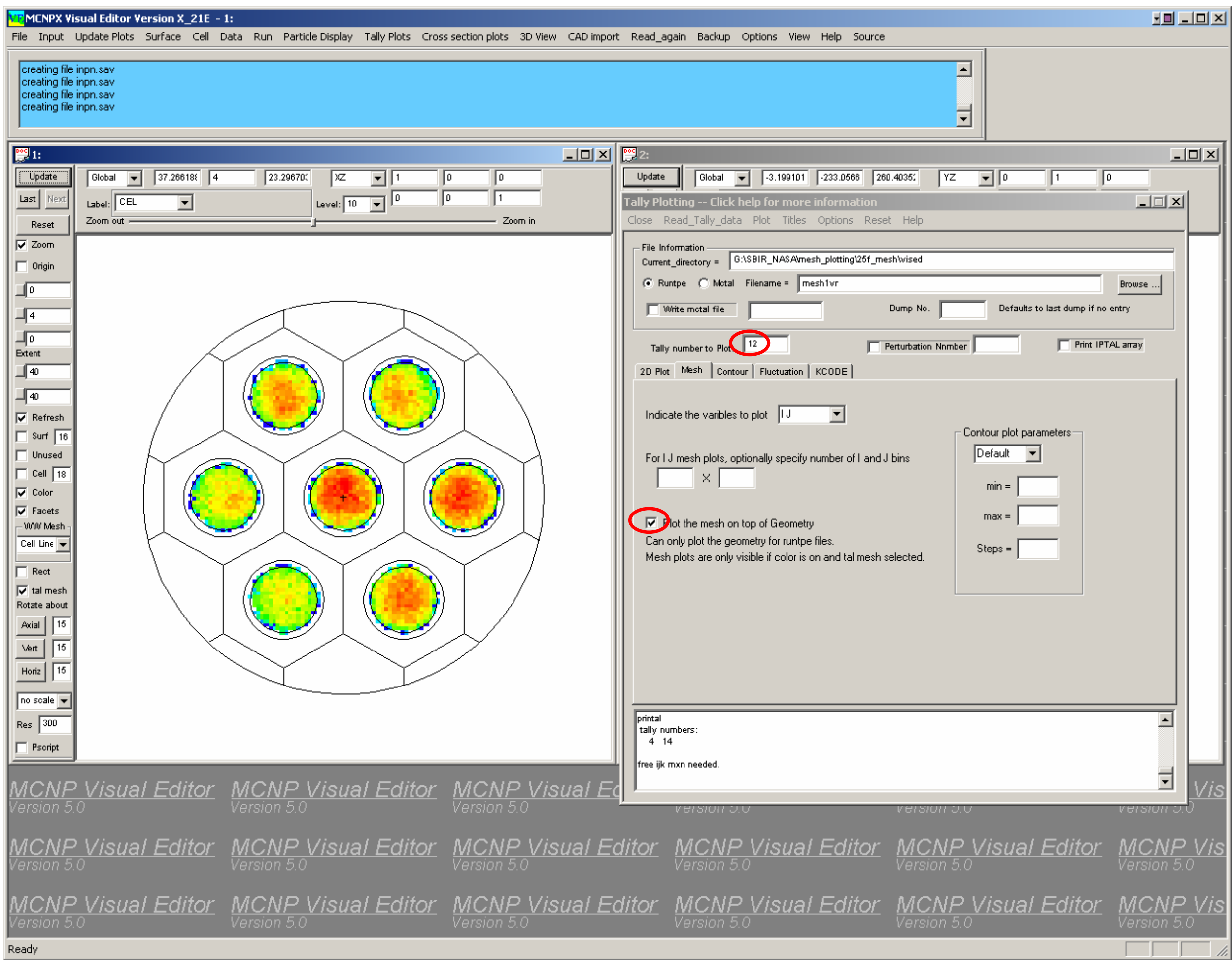

Figure 24. Mesh Tally Plot on Top of the Geometry.

Once the plot has been generated, the view can be changed to show different regions of the tally mesh. In Figure 25 the zoom toggle button was selected and the mouse was dragged across one of the barrels to zoom in on a single barrel.

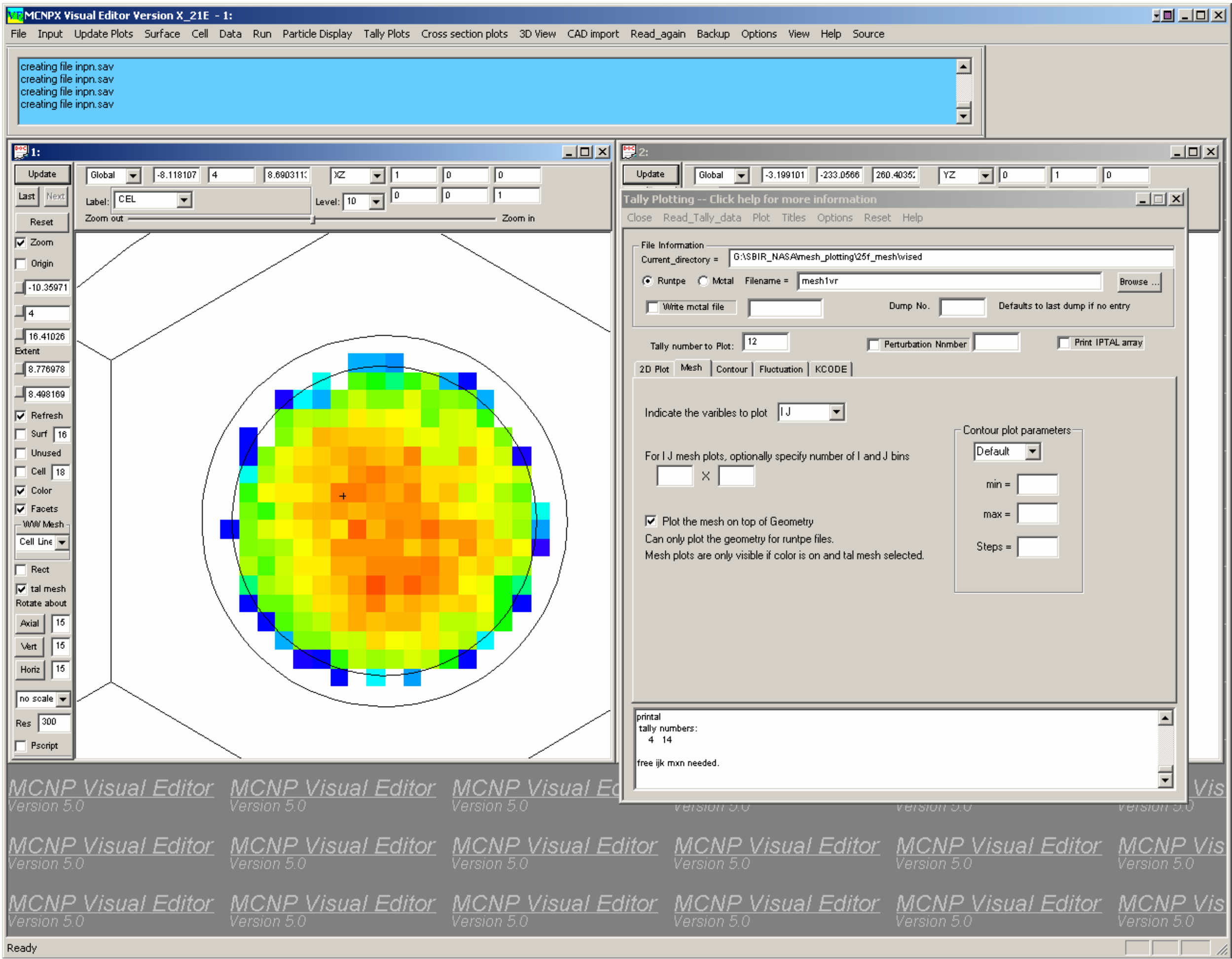

Figure 25. Zoomed in Region of the Mesh Tally Plot.

The latest MCNPX beta version has a number of mesh test problems contained in the testmesh directory that is downloadable from the MCNPX beta site. An attempt was made to reproduce the test plots in both the MCNPX plotter and in the Visual Editor. The following figures show plots made both in the MCNPX tally plotter and the Visual Editor.

Figure 26 shows a contour plot generated with the MCNPX tally plotter using the command "rmc inp01m tal 5 free sc contour line". Figure 27 shows the same plot generated in the MCPNX Visual Editor.

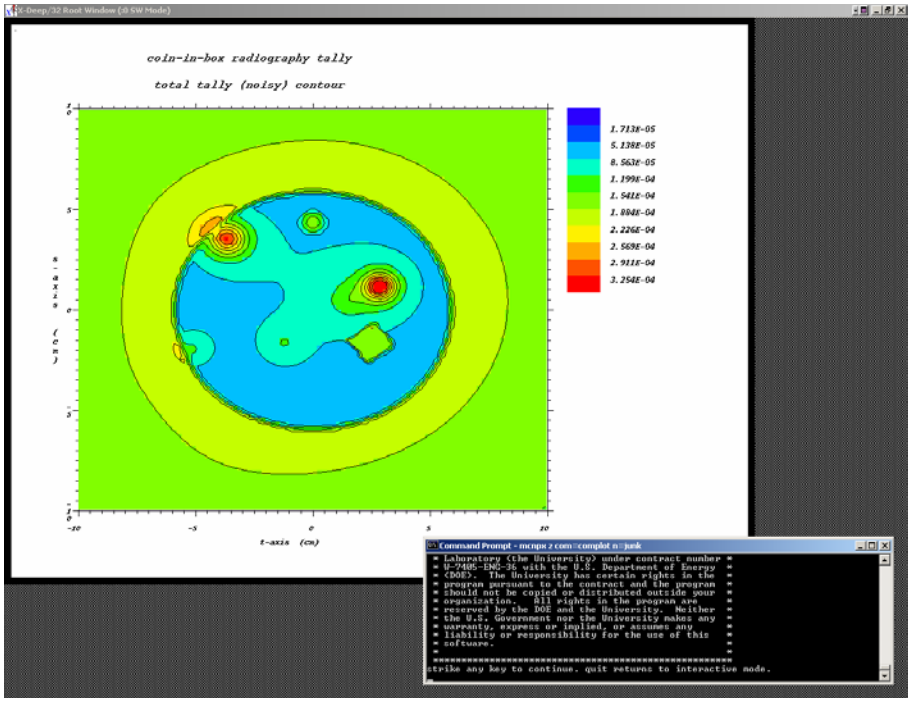

Figure 26. MCNPX Contour Plot.

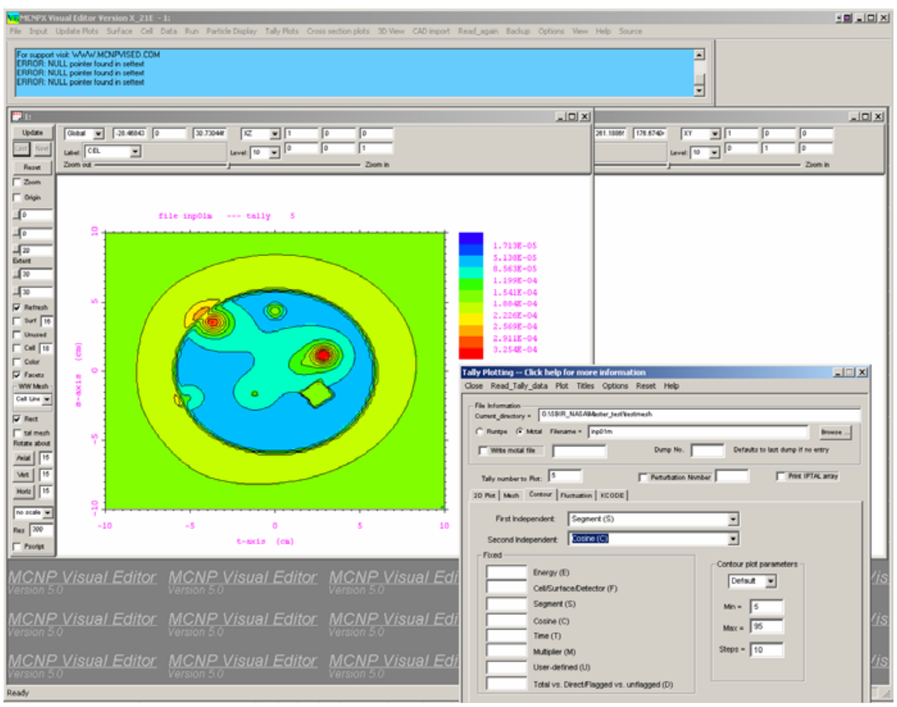

Figure 27. Visual Editor Contour Plot

Figure 28 shows a contour plot generated with the MCNPX tally plotter using the command “rmc inp01m tal 5 free sc contour line fixed d 2”. Figure 29 shows the same plot generated in the MCPNX Visual Editor.

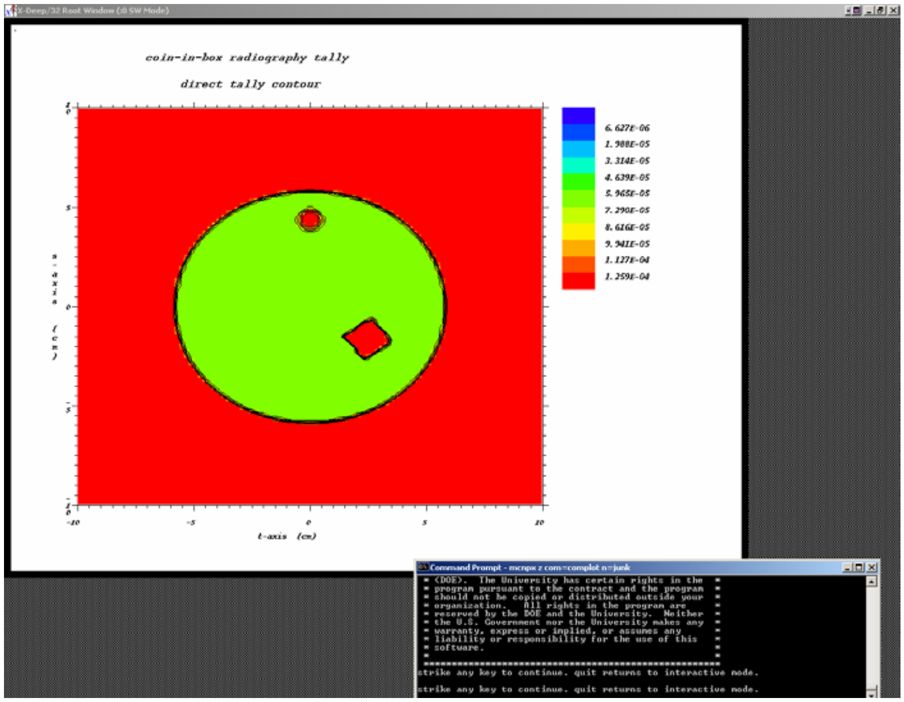

Figure 28. MCNPX Contour Plot with a Fixed Variable.

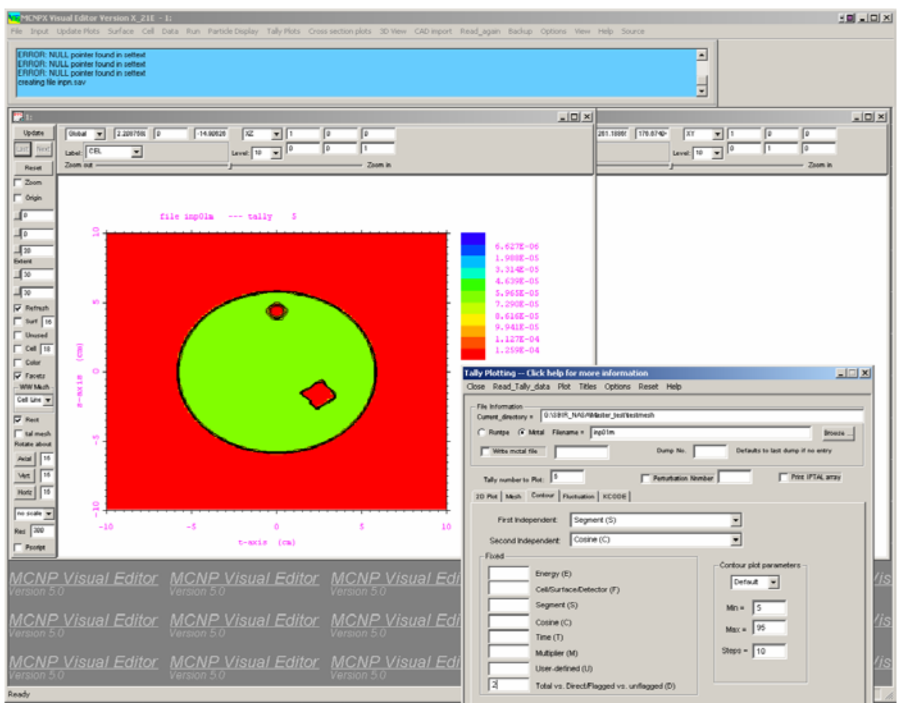

Figure 29. Visual Editor Contour Plot with a Fixed Variable.

Figure 30 shows a contour plot for testmesh input file 4 generated with the MCNPX tally plotter using the command "tal 3 free ik". Figure 31 shows the same plot generated in the MCPNX Visual Editor.

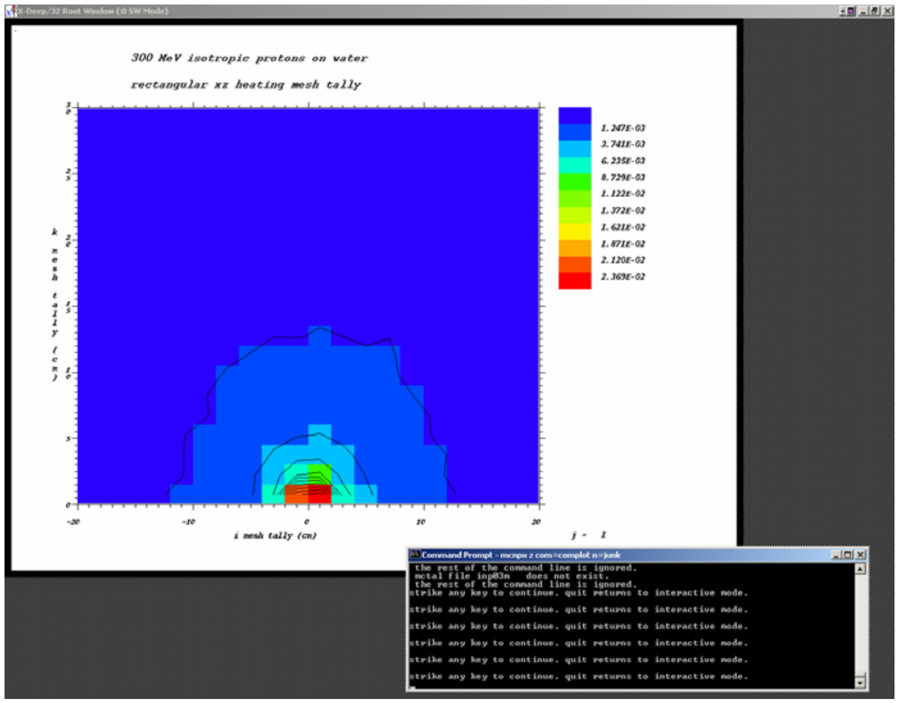

Figure 30. MCNPX Mesh Tally Plot.

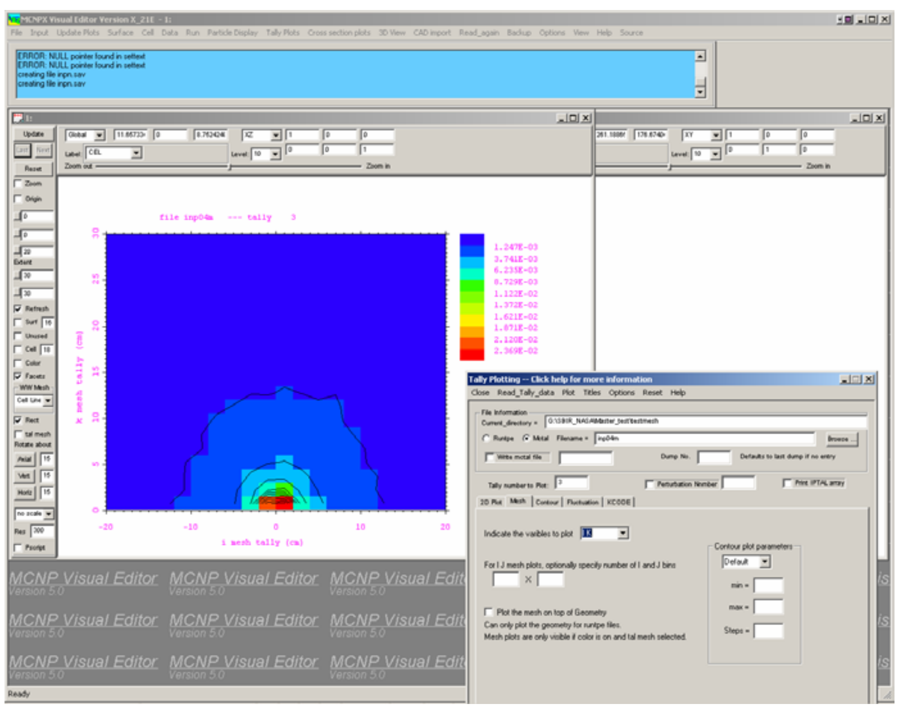

Figure 31. Visual Editor Mesh Tally Plot.

Figure 32 shows a contour plot for testmesh input file 4 generated with the MCNPX tally plotter using the command "tal 1 free jk contour log". Figure 33 shows the same plot generated in the MCPNX Visual Editor.

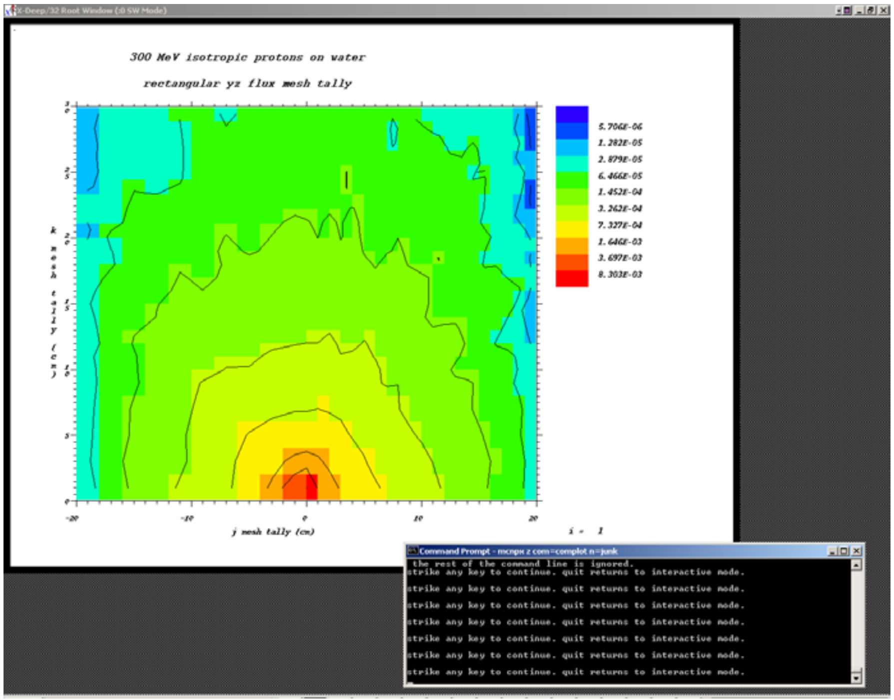

Figure 32. MCNPX Mesh Tally Plot.

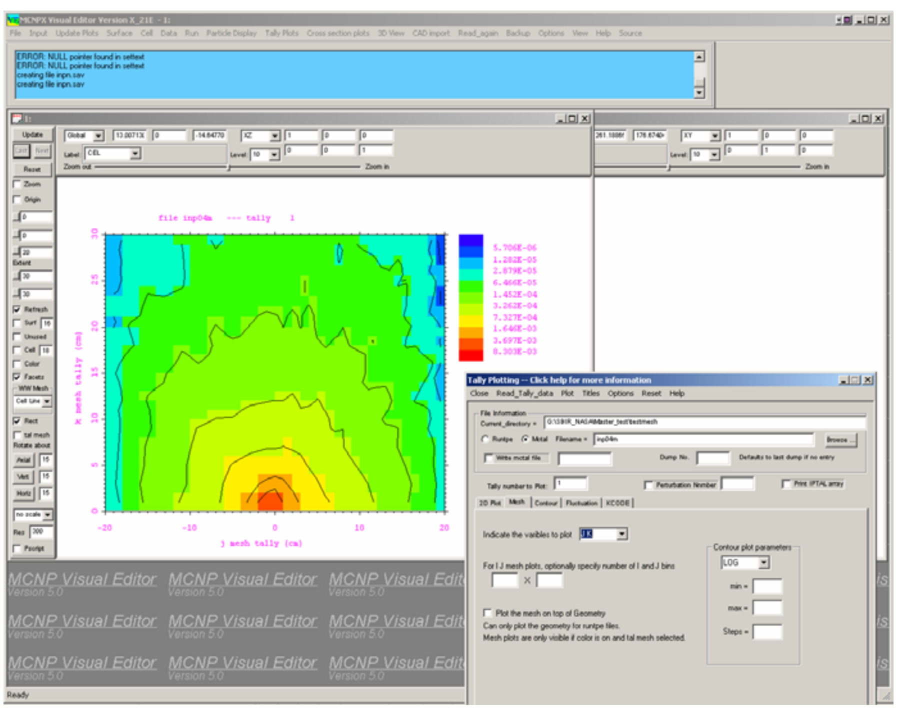

Figure 33. Visual Editor Mesh Tally Plot.

Figure 34 shows a contour plot for testmesh input file 4 showing the same mesh in cylindrical profile generated with the MCNPX tally plotter using the command "tal 11 free ij contour 5e-5 .02 20 log". Figure 35 shows the same plot generated in the MCPNX Visual Editor.

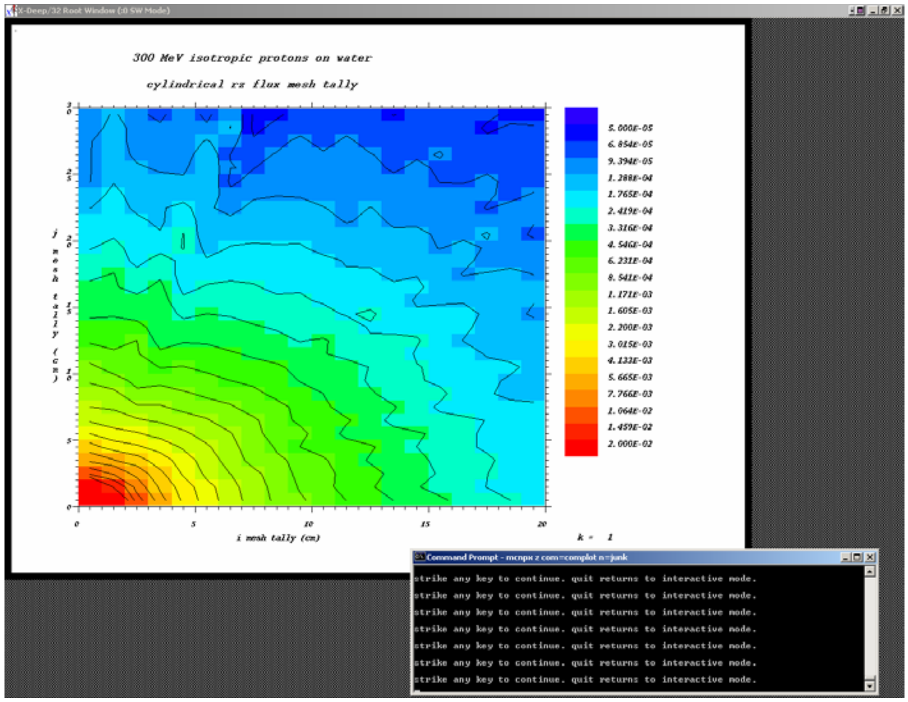

Figure 34. MCNPX Mesh Tally Plot.

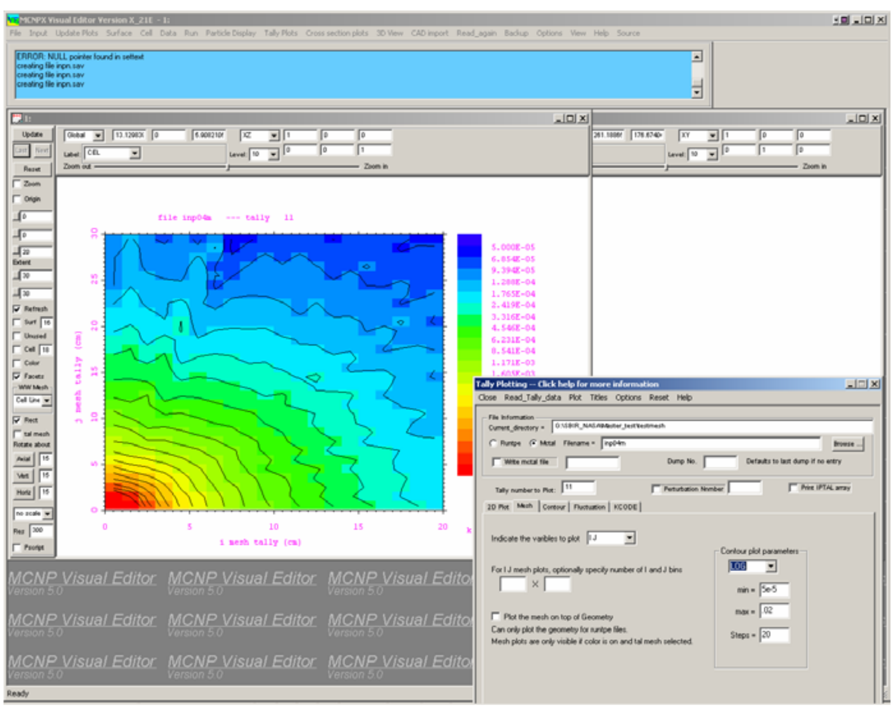

Figure 35. Visual Editor Mesh Tally Plot.

Figure 36 shows a contour plot for testmesh input file 5 generated with the MCNPX tally plotter using the command “rmc inp05m tal 1 free jk contour 5 95 10 %”. Figure 37 shows the same plot generated in the MCPNX Visual Editor.

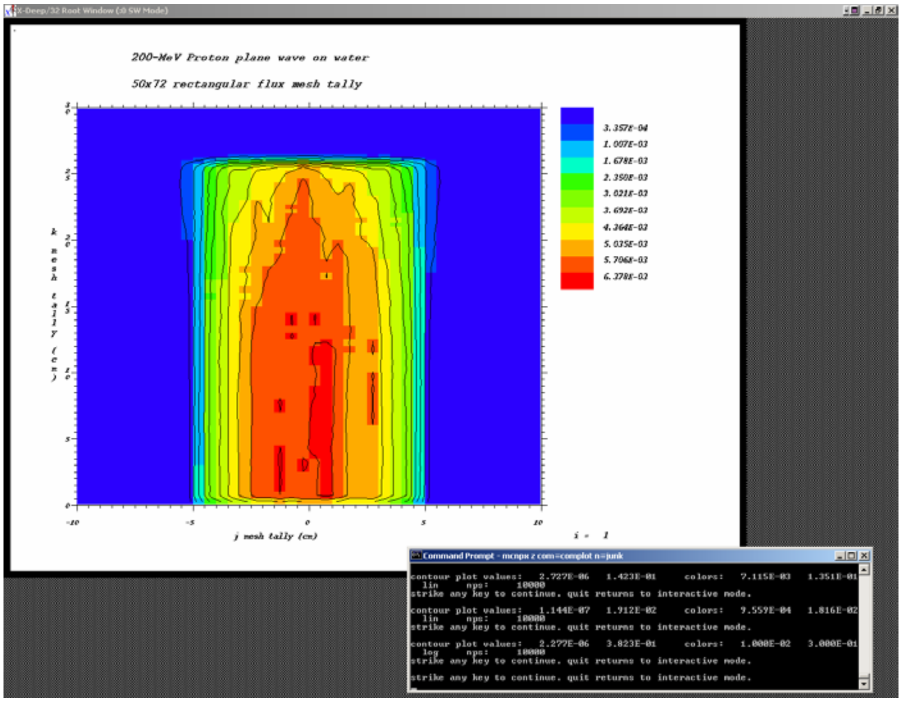

Figure 36. MCNPX Mesh Tally Plot.

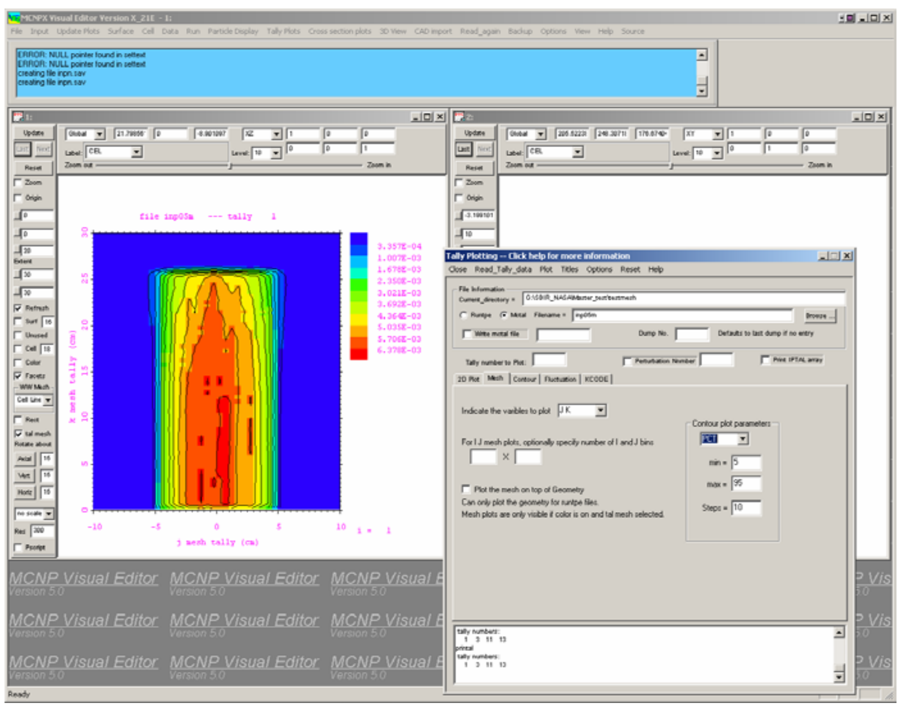

Figure 37. Visual Editor Mesh Tally Plot.

Figure 38 shows a contour plot for testmesh input file 4 generated with the MCNPX tally plotter using the command "tal 13 free ij contour .01 .3 20 log". Figure 39 shows the same plot generated in the MCPNX Visual Editor.

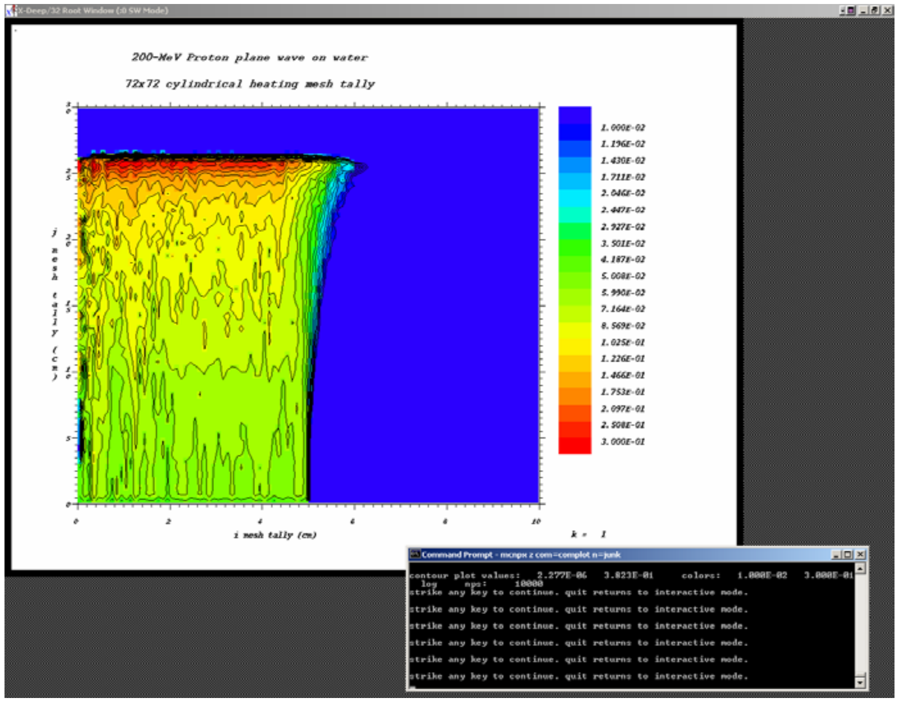

Figure 38. MCNPX Mesh Tally Plot.

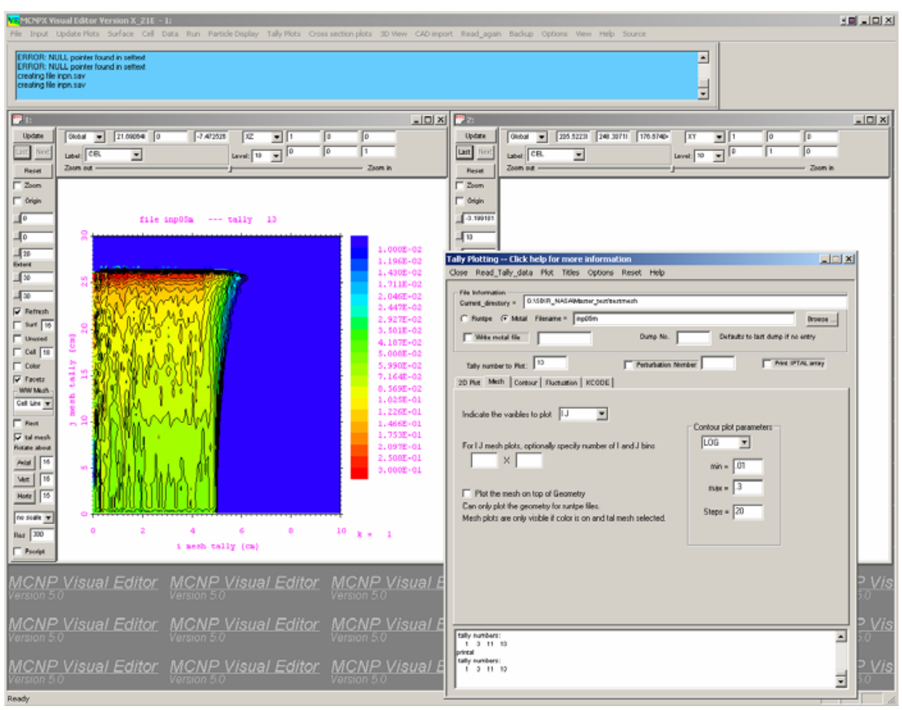

Figure 39. Visual Editor Mesh Tally Plot.

Figure 40 shows a contour plot for testmesh input file 10 generated with the MCNPX tally plotter using the command “rmc inp10m tal 2 free jk con 1e-5 .05 20 log”. Figure 41 shows the same plot generated in the MCPNX Visual Editor.

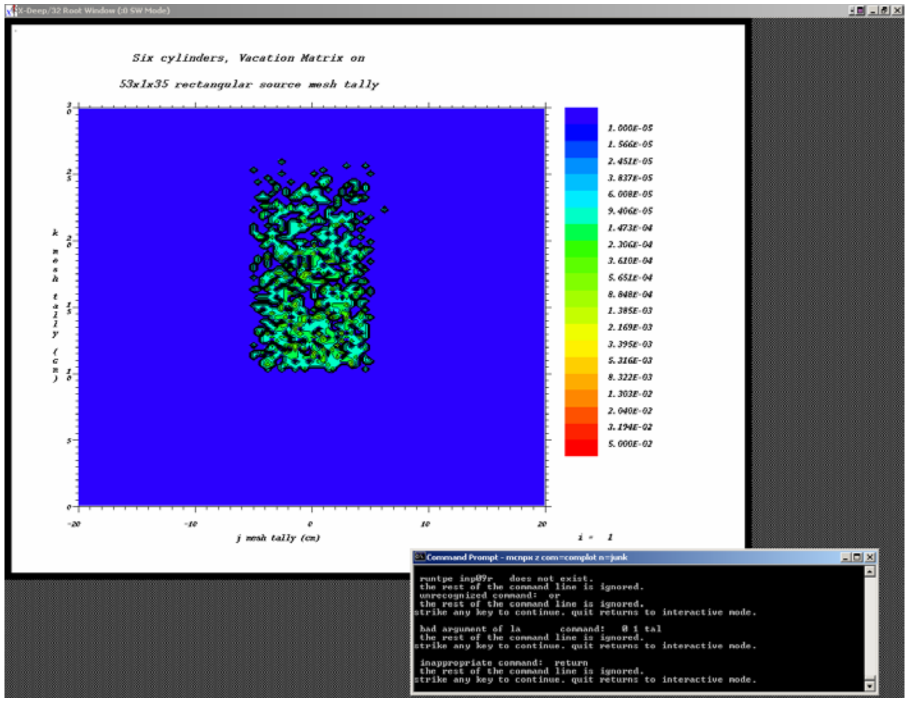

Figure 40. MCNPX Mesh Tally Plot.

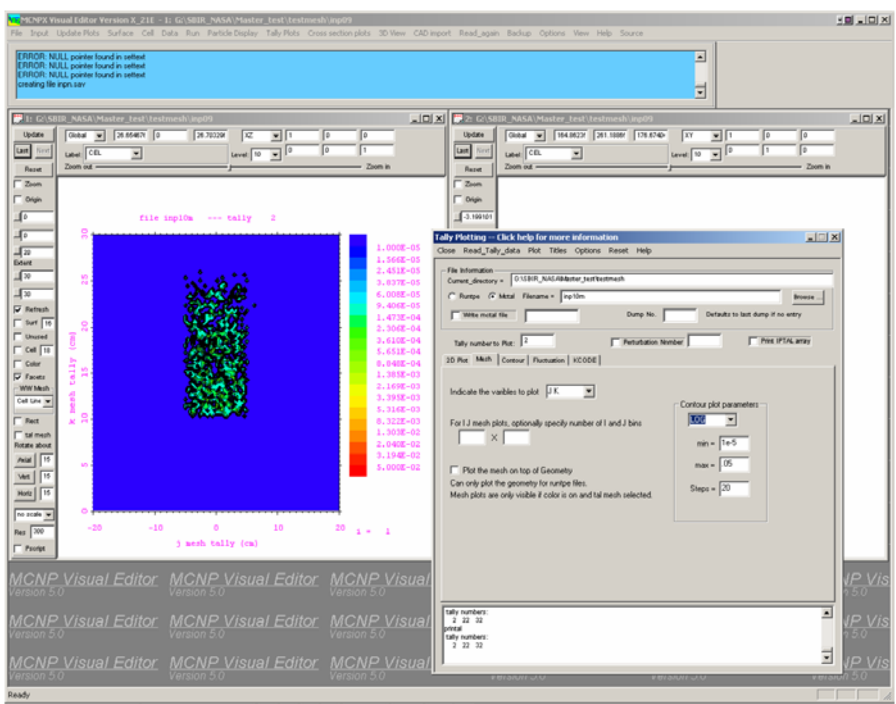

Figure 41. Visual Editor Mesh Tally Plot.

Figure 42 shows an expanded view of the previous tally plot within the MCNPX Visual Editor.

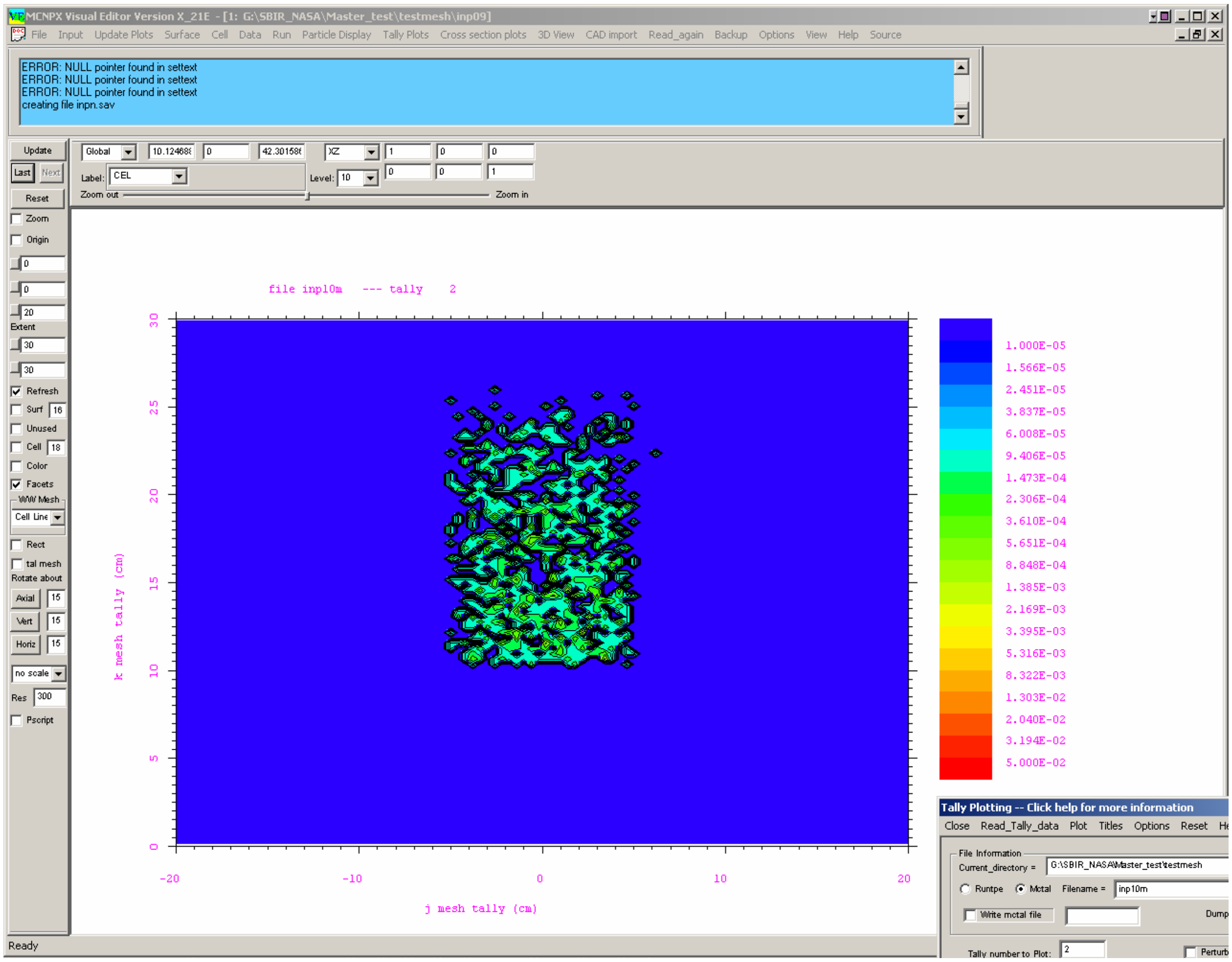

Figure 42. Enlarged View of a Visual Editor Mesh Tally Plot.