

Visualization of Output Data from Particle Transport Codes

Phase 1 Final Report

Instrument No: DE-FG02-07ER84735

June 20, 2007 through March 19, 2008

Recipient: Randolph Schwarz, Visual Editor Consultants

Principal Investigator: Randolph Schwarz

Teaming members

Randolph Schwarz of Visual Editor Consultants

Leland L. Carter of Carter Monte Carlo Analysis, Inc.

Alysia Schwarz of Visual Editor Consultant

Table of Contents

	<u>PAGE</u>
TABLE OF FIGURES.....	III
1 EXECUTIVE SUMMARY	1
2 SIGNIFICANT ACCOMPLISHMENTS	1
2.1 DISPLAYING OUTPUT DATA ON 2D GEOMETRY PLOTS.....	3
2.2 DISPLAYING PARTICLE DATA	5
2.3 DISPLAYING TALLY DATA	18
2.4 EXPORTING OUTPUT DATA.....	29
2.5 ADVANCED DISPLAY OPTIONS.....	31
3 FUTURE WORK.....	32
3.1 EXPORTING MCNP OUTPUT DATA.....	32
3.2 ADVANCED TALLY OPTIONS.....	32
3.3 ADVANCED DISPLAY OPTIONS.....	32

Table of Figures

	<u>Page</u>
Figure 2.1. Cask Geometry Displayed from the MCNP Dump File.....	3
Figure 2.2. Color by Particle Activity in Each Cell.....	4
Figure 2.3. Enhanced Particle Track Plotting Panel.....	5
Figure 2.4. Plotting the Proton Plane Source.....	7
Figure 2.5. Plotting Collision Points for a Multiple Particle Problem.....	8
Figure 2.6. Collision Points Showing Different Particle Types.....	9
Figure 2.7. Collisions with Neutrons and Photons Turned Off.....	10
Figure 2.8. Collisions with Proton Source Turned Off.....	11
Figure 2.9. Particle Tracks for Different Particle Types.....	12
Figure 2.10. Carbon Beam Therapy Problem.....	13
Figure 2.11. Carbon Beam Showing Secondary Protons and Neutrons.....	14
Figure 2.12. Iron Ions on a Water Phantom.....	15
Figure 2.13. Alpha Particles Generated from Iron Ions on a Water Phantom.....	16
Figure 2.14. Uranium Isotopes on Lithium.....	17
Figure 2.15. Mesh Tally Plot in the MCNPX Visual Editor.....	18
Figure 2.16. Mesh Tally Plot on Top of the Geometry.....	19
Figure 2.17. Zoomed in Region of the Mesh Tally Plot.....	20
Figure 2.18. MCNPX Contour Plot.....	21
Figure 2.19. Visual Editor Contour Plot.....	21
Figure 2.20. MCNPX Contour Plot with a Fixed Variable.....	22
Figure 2.21. Visual Editor Contour Plot with a Fixed Variable.....	22
Figure 2.22. MCNPX Mesh Tally Plot.....	23
Figure 2.23. Visual Editor Mesh Tally Plot.....	23
Figure 2.24. MCNPX Mesh Tally Plot.....	24
Figure 2.25. Visual Editor Mesh Tally Plot.....	24
Figure 2.26. MCNPX Mesh Tally Plot.....	25
Figure 2.27. Visual Editor Mesh Tally Plot.....	25
Figure 2.28. MCNPX Mesh Tally Plot.....	26
Figure 2.29. Visual Editor Mesh Tally Plot.....	26
Figure 2.30. MCNPX Mesh Tally Plot.....	27
Figure 2.31. Visual Editor Mesh Tally Plot.....	27
Figure 2.32. MCNPX Mesh Tally Plot.....	28
Figure 2.33. Visual Editor Mesh Tally Plot.....	28
Figure 2.34. Exporting Table 126 Data.....	29
Figure 2.35. Table 126 Data Imported into a Spreadsheet.....	30
Figure 2.36. Color 3D View of Cell Importances.....	31
Figure 3.1. 3D Display of Collision Points.....	33

1 Executive Summary

The Monte Carlo N-Particle (MCNP) and Monte Carl N-Particle eXtended (MCNPX) computer codes are used extensively around the world for particle transport calculations. A tremendous amount of data are generated in running these complex particle transport codes. These data include detailed information about the transport of the particles and the results. Few options are currently available for displaying the MCNP results visually in a manner that is easily understandable.

Programs exist to display tally data, but it is hard to do a complete visual interrogation of the final calculated results. This information is typically contained in a binary file after the run finishes (i.e. runtpe) and the output file. The MCNP Visual Editor will be enhanced so that it can read these files and use graphical displays to show important transport data to the user.

The visualization of the output was investigated and a post-processing visualization tool was developed. This provides the user a useful tool to analyze the calculational results and provide insight into options for improving the results by graphically displaying the important information generated during the run.

All of these new capabilities were added to the MCNP Visual editor and will be eventually be distributed through the Radiation Safety Information Computational Center (RSICC) in Oak Ridge.

2 Significant Accomplishments

This work focused specifically on the MCNPX code because it has more capabilities than MCNP5, including multiple mesh tallies and multiple particle types. Because LANL has indicated that MCNP5 and MCNPX will be combined in the future, it is essential that the additional features of MCNPX be supported. Once the MCNPX features have been addressed it is possible to then generate an MCNP5 version by removing the options not available in MCNP5. In the discussion that follows, MCNP is meant to generically refer to both the MCNP5 and MCNPX code and the eventual MCNP6 code that is expected to be generated that combines both codes, because the capabilities developed here will be adaptable to these other versions of the code.

It was decided that the Visual Editor was the best choice for displaying output data because of its widespread use for displaying MCNP geometries. The MCNP plotter is capable of performing many of the 2D displays, but the MCNP plotter requires an external X windows server be used on a Windows platform and some of the user options are hard to understand, especially when it comes to plotting data from the MCNP dump file. Once the Visual Editor was able to read in the 2D data, it was then adaptable for generating more complex 3D plots of the data and exporting particle activity to an external file. For these advanced features to work, however, the Visual Editor must first be able to read the MCNP binary dump file.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

The initial effort focused on reading an MCNP binary dump file (i.e. runtpe) and then displaying 2D plots of the MCNP data on top of the MCNP geometry (see section 2.1). Additional work was performed to display particle tracks for multiple particle types (see section 2.2) and to also display mesh tallies (see section 2.3). Work was also done to read in an output data table and write it out in a format that can be easily read by Excel© (see section 2.4). Advanced features were also investigated, including the generation of 3D geometry plots with the cells colored according to user specified data parameters (see section 2.5).

2.1 Displaying Output Data on 2D Geometry Plots

One of the goals of this work was to demonstrate that a MCNP binary dump file could be read in and the associated data displayed in the Visual Editor. This will lay the ground work for more advanced geometry displays in the Visual Editor.

Figure 2.1 shows an MCNP geometry that consists of sixteen cesium chloride capsules. The source is a photon source generated from the decay products of the cesium. There is a hole in the cask, shown in the left plot, and the purpose of the calculation is to calculate the dose rate out the hole. This plot was created after successfully reading in the MCNP binary dump file.

Included in Figure 2.1 is the panel that has been added for reading in the dump files. The user brings up this new panel by selecting the “Tally Plots->Read Runtpe Data” menu option. The user then indicates the binary dump file to read and then selects “Read_runtpe_file”. This will read in the MCNP dump file, after which the user can select “Update Plots” to plot the geometry.

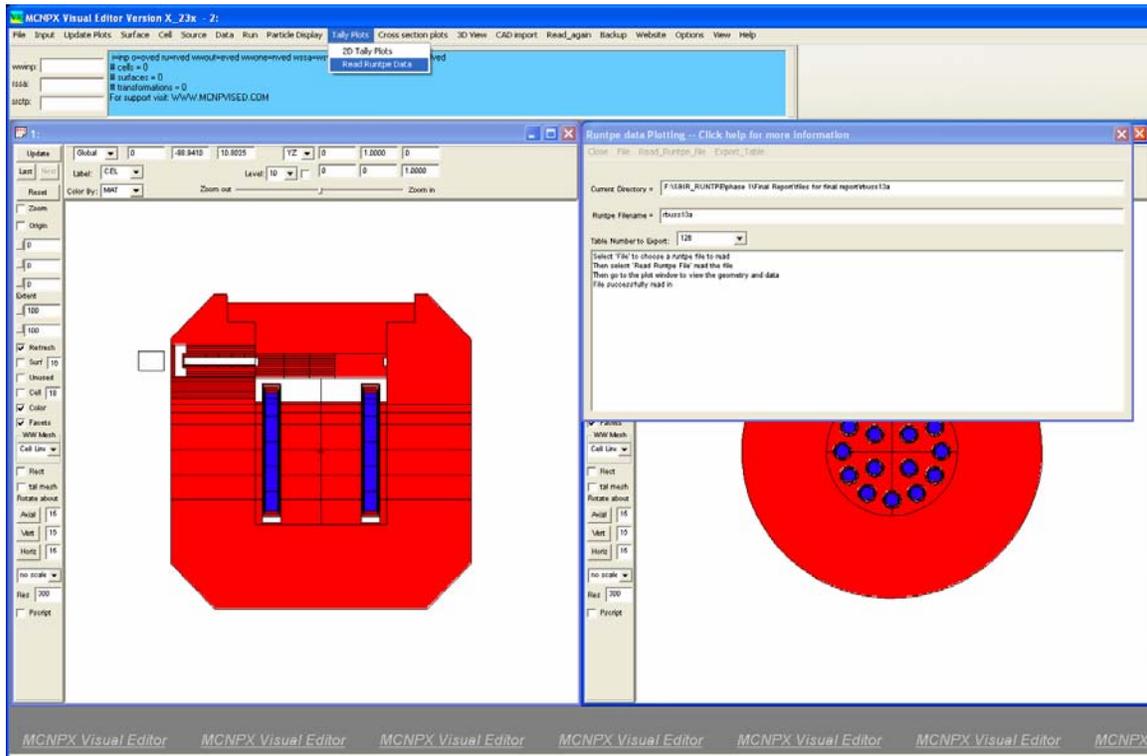


Figure 2.1. Cask Geometry Displayed from the MCNP Dump File.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Once the MCNP dump file has been read in, information concerning the running of the code can be plotted. Of particular interest is the particle activity in each cell, which is detailed in print table 126 of the output file. This table consists of eight columns of data for each cell in the input file and includes information concerning the particle activity in each cell for each particle type in the problem.

It is hard to get an understanding of these numbers by looking at them, but the Visual Editor now allows the user to plot the activity values and to color the cells by these particle activity values. Figure 2.2 shows the colors for the number of photon tracks entering each cell in the left plot window and the colors for the photon “collisions multiplied by weight” in each cell in the right hand window. By using color the user can immediately identify problem areas. In Figure 2.2 the thin shield at the end of the hole is cyan in the left plot, indicating that not many particles have reached this important region. In the right plot, this problem is further highlighted. The blue has a collision*weight contribution on the order of 10^7 , while cells further down the hole are on the order of 10^{10} . This decrease can be observed in the dark blue color in the cells at the end of the port.

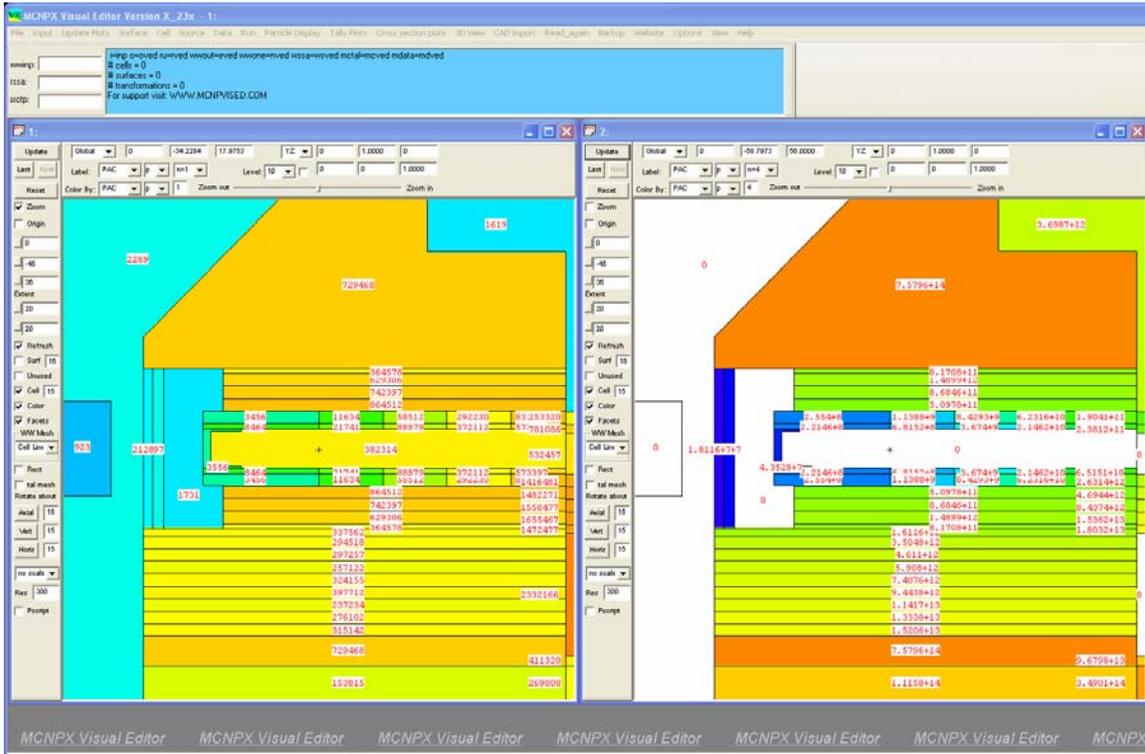


Figure 2.2. Color by Particle Activity in Each Cell.

The user can plot the particle information data and colors for each of the eight columns listed in table 126 of the output file. The available data include 1) tracks entering; 2) population; 3) collisions; 4) collisions * weight; 5) number weighted energy; 6) flux weighted energy; 7) average track weight; 8) average track mfp.

This work was essential to form a basis for the future work that may be done to generate 3D plots of this same information

2.2 Displaying Particle Data

Collision point plotting has been available in the Visual Editor for a number of years. As part of this effort, this capability was updated to allow for the multiple particle types available in MCNPX and to give the user many options for displaying the particles.

Figure 2.3 shows the upgraded particle track-plotting panel. This panel can be accessed by selecting Particle Display->Plot Particle Tracks from the main menu.

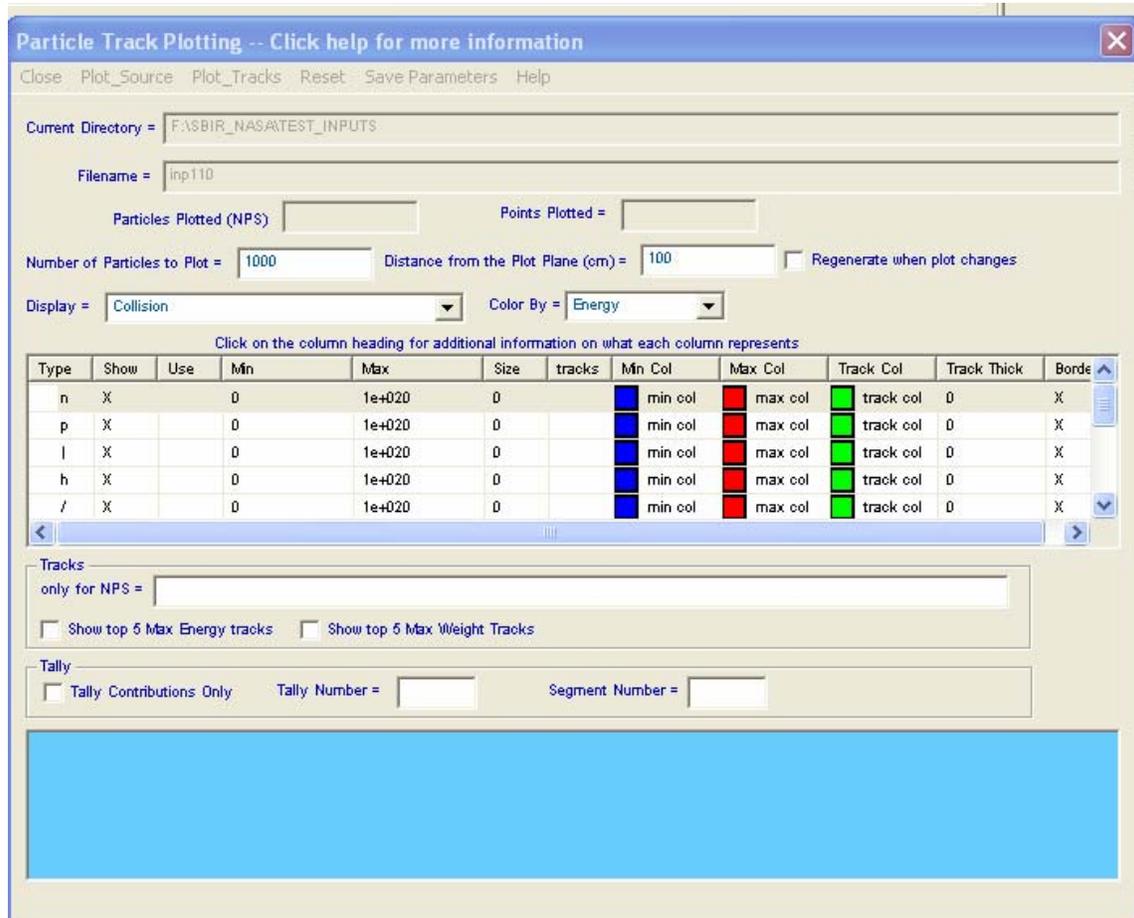


Figure 2.3. Enhanced Particle Track Plotting Panel.

Particle track plotting in the Visual Editor works by having the Fortran MCNP 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 2.3, test input file inp110 has been read in. Ten particle types are identified in the mode card:

```
mode n p | h / z d t s a
```

This mode card specifies neutrons (n), photons (p), muons (l), protons (h), pions (/), neutral pions (z), deuterons (d), tritons (t), helium-3 (s), and helium-4 (a). The source is a surface source incident on a cylindrical iron target.

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”. This column indicates the min and max values, as specified in the following two columns, should 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 and plot all particles.

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.

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.

Phase 1 Final Report
 Visualization of Output Data from Particle Transport Codes

In Figure 2.4, 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 2.3. The source definition is defined as follows:

```
sdef sur = 1 erg = 800 par = h dir = 1 pos = 0 0 0 rad = d1
sil 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 “sil”. 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. In Figure 2.4, the size of the Proton source particle has been changed from the default of 0 to 3 so the points can be seen.

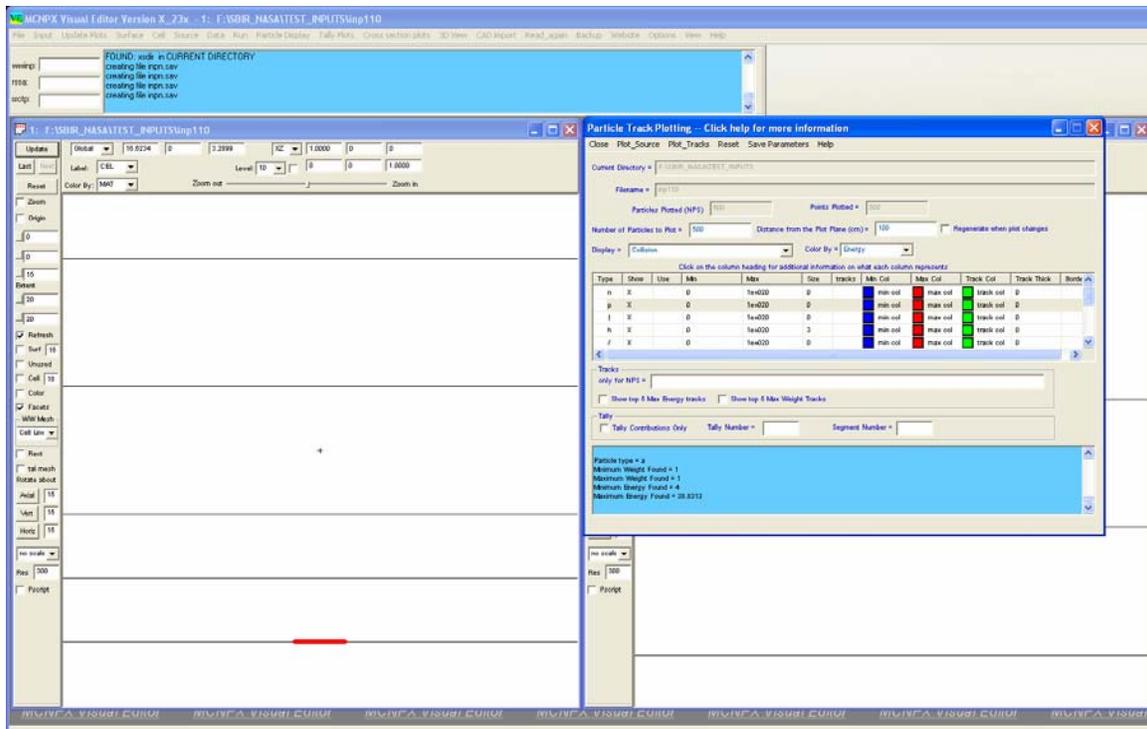


Figure 2.4. Plotting the Proton Plane Source.

Phase 1 Final Report
 Visualization of Output Data from Particle Transport Codes

In Figure 2.5, “Plot tracks” is selected to plot the particle tracks. Because the source particle (protons) is enlarged (set to size 3), 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. Because all of the secondary particles are the same size and color it is hard to differentiate the type of particle being generated. The border surrounding each particle has been removed to accentuate the colors.

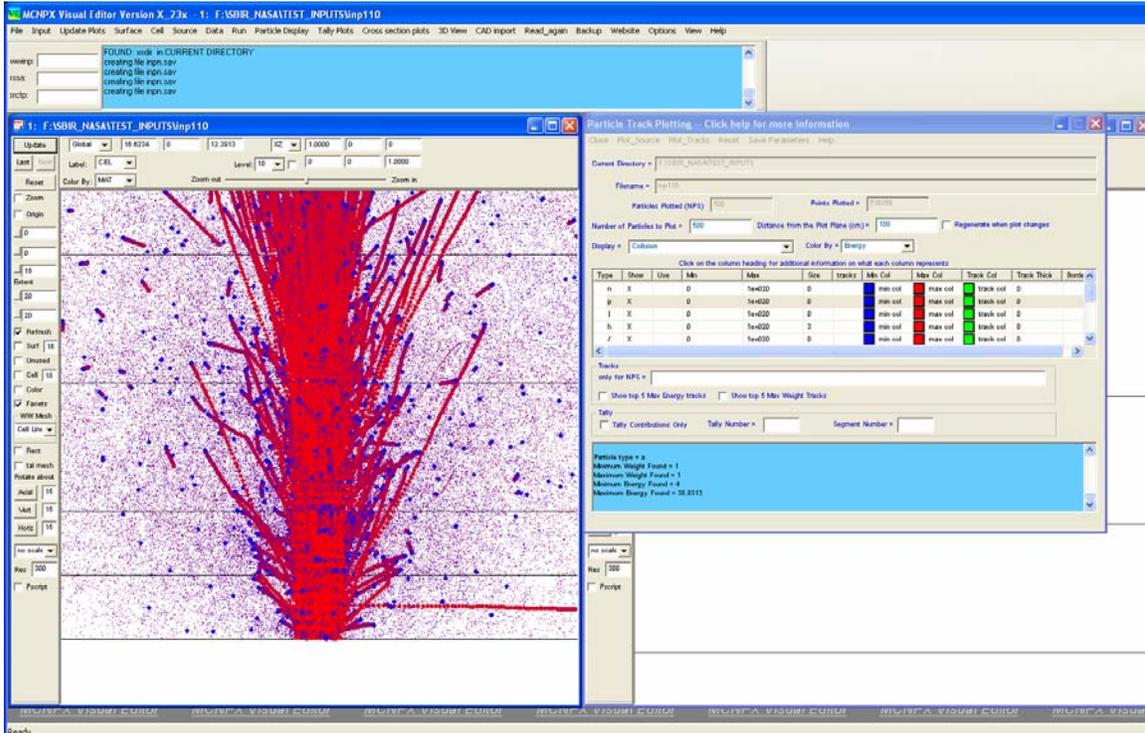


Figure 2.5. Plotting Collision Points for a Multiple Particle Problem.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

In Figure 2.6 the neutrons and photons have been set to the smallest size and are colored yellow and green so they are not as noticeable. To differentiate the different particles that are generated, the remaining particles are given a size of four (4) and each is colored with its own unique color using the “color picker”. A total of 500 source particles were used to generate this plot.

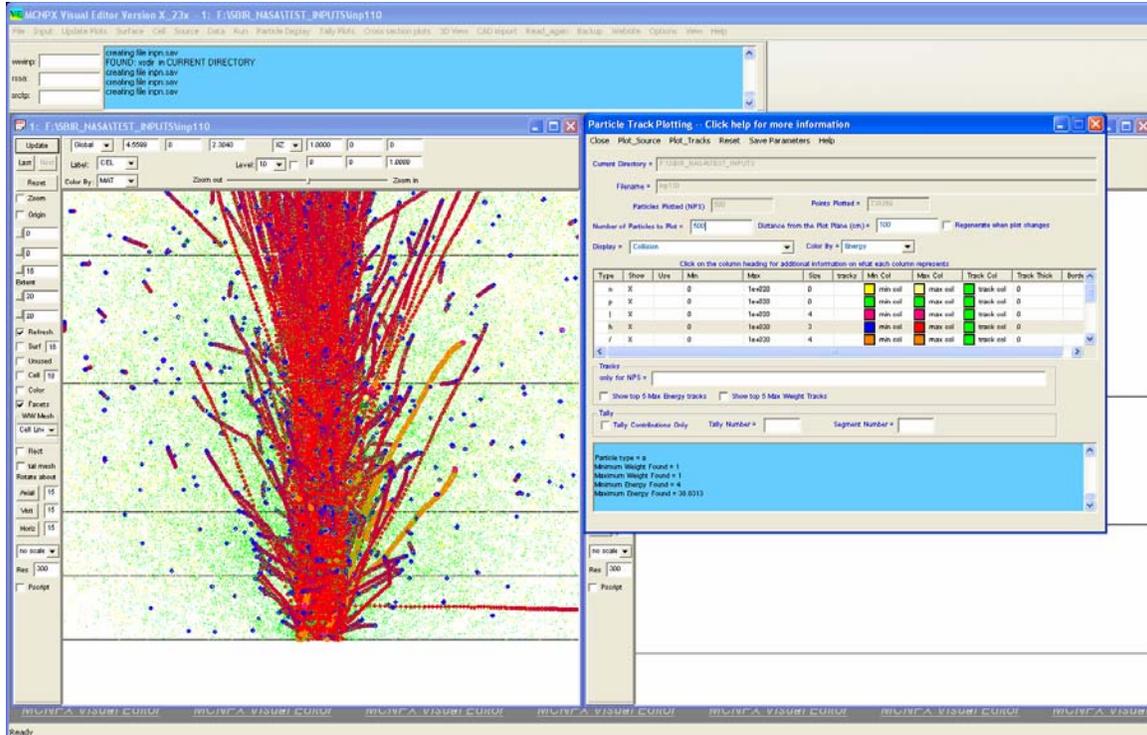


Figure 2.6. Collision Points Showing Different Particle Types.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

In Figure 2.7, the plotting of the neutrons and photons has been turned off, and the size of the source protons has been set to the smallest size to further highlight the particles that are generated.

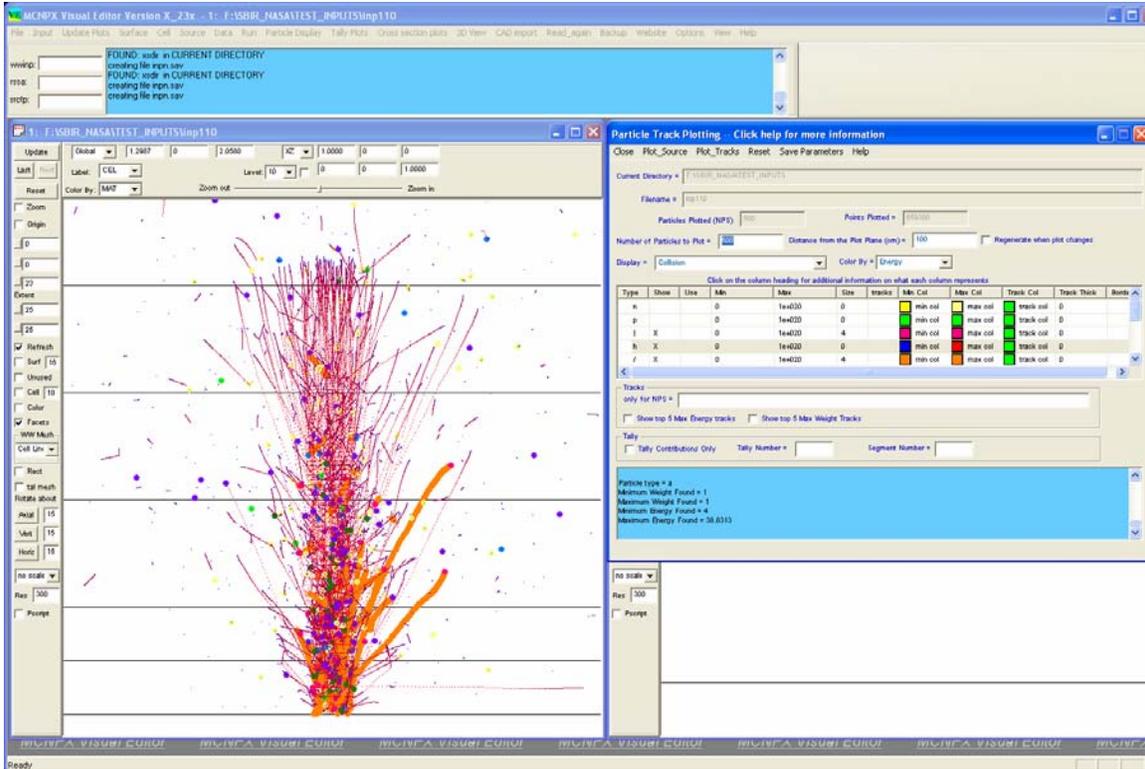


Figure 2.7. Collisions with Neutrons and Photons Turned Off.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

In Figure 2.8, the proton source has been turned off and the pion collisions have been set to a size one (1) to highlight the remaining particles. It can be observed (by noticing the user specified size and color of each particle) that the generated pions have a number of close collisions ending in the creation of a muon. By being able to visualize the collisions points, the user can gain valuable insight into the physics of the problem. The user can control the color, size, and visibility of each particle type. This allows the removal of uninteresting particles so the user can focus in on the particles that are important for the problem.

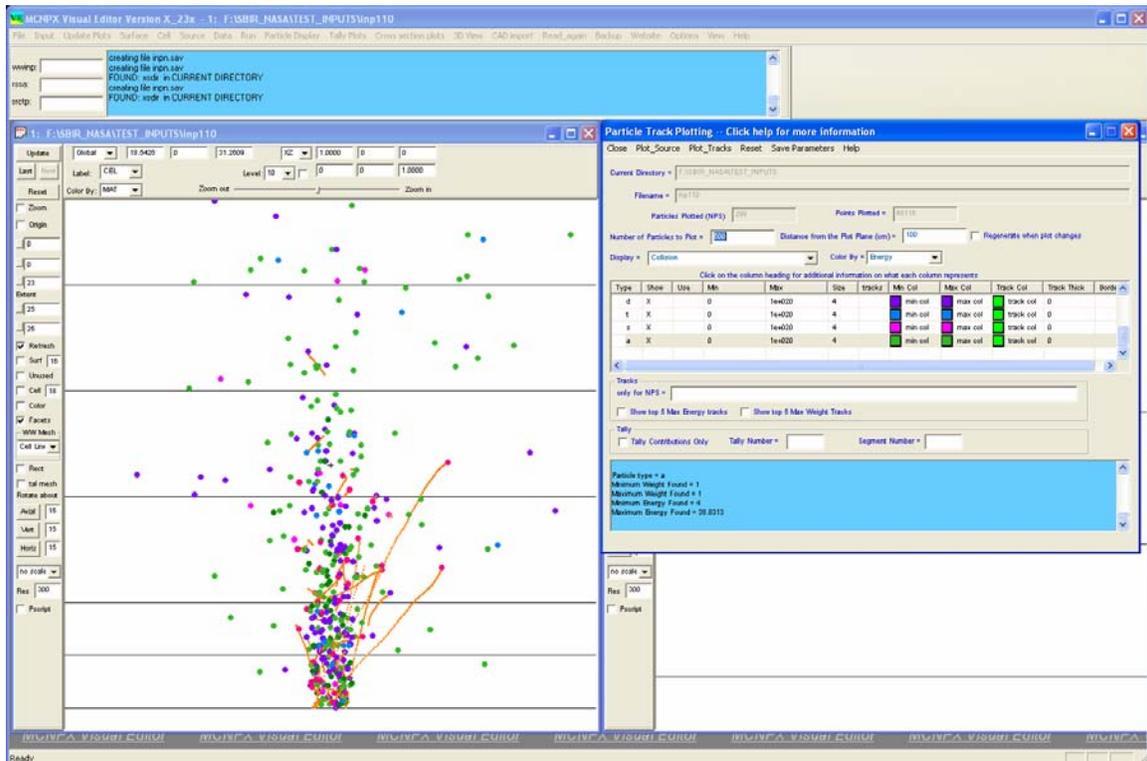


Figure 2.8. Collisions with Proton Source Turned Off.

Phase 1 Final Report
 Visualization of Output Data from Particle Transport Codes

In Figure 2.9 particle tracks have been turned on for the different particle types. The beginning of the particle track can be identified with an open circle. The color of the track has been set to the same color as the particle type. This feature can be used to provide additional insight on the propagation of particles in the problem.

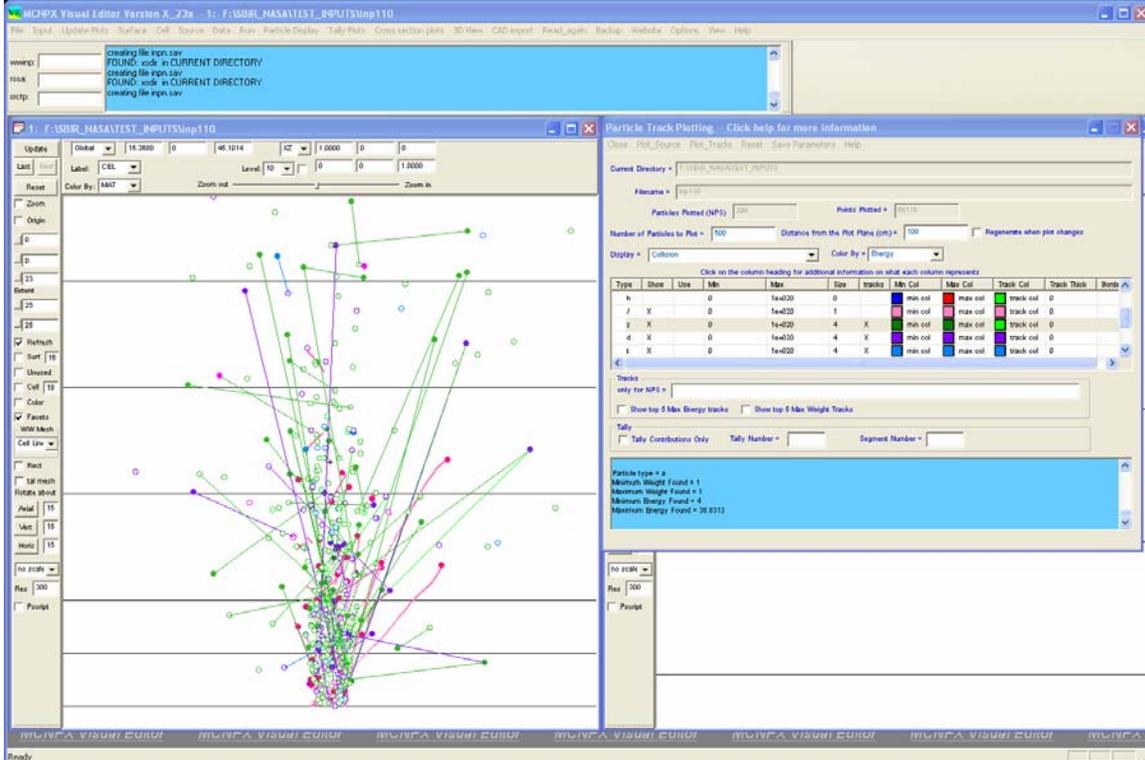


Figure 2.9. Particle Tracks for Different Particle Types.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.10 shows a particle track plot of a carbon beam therapy problem. Carbon nuclei come in from the top of the geometry. The carbon beam stops at the tumor region in the center of the sphere. Notice that the carbon beam loses energy in the tumor region as shown by the particle beam turning blue.

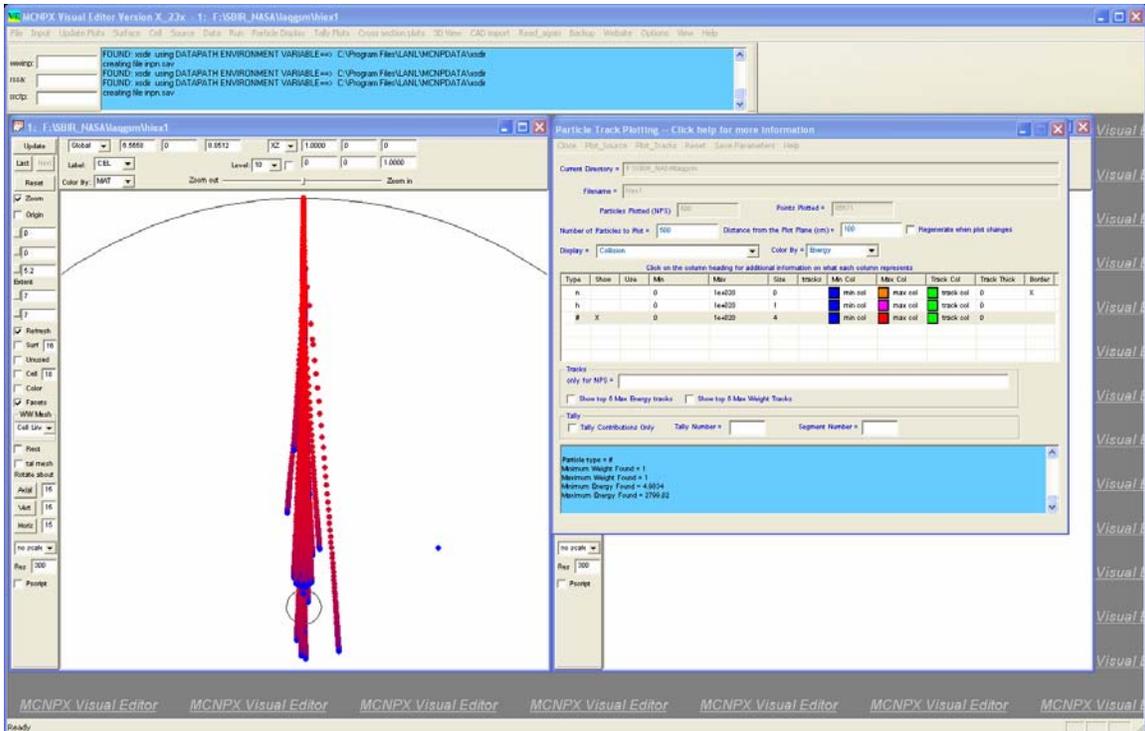


Figure 2.10. Carbon Beam Therapy Problem.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.11 shows the secondary particles produced by the carbon beam. Protons are shown in purple and secondary neutrons are shown in light green.

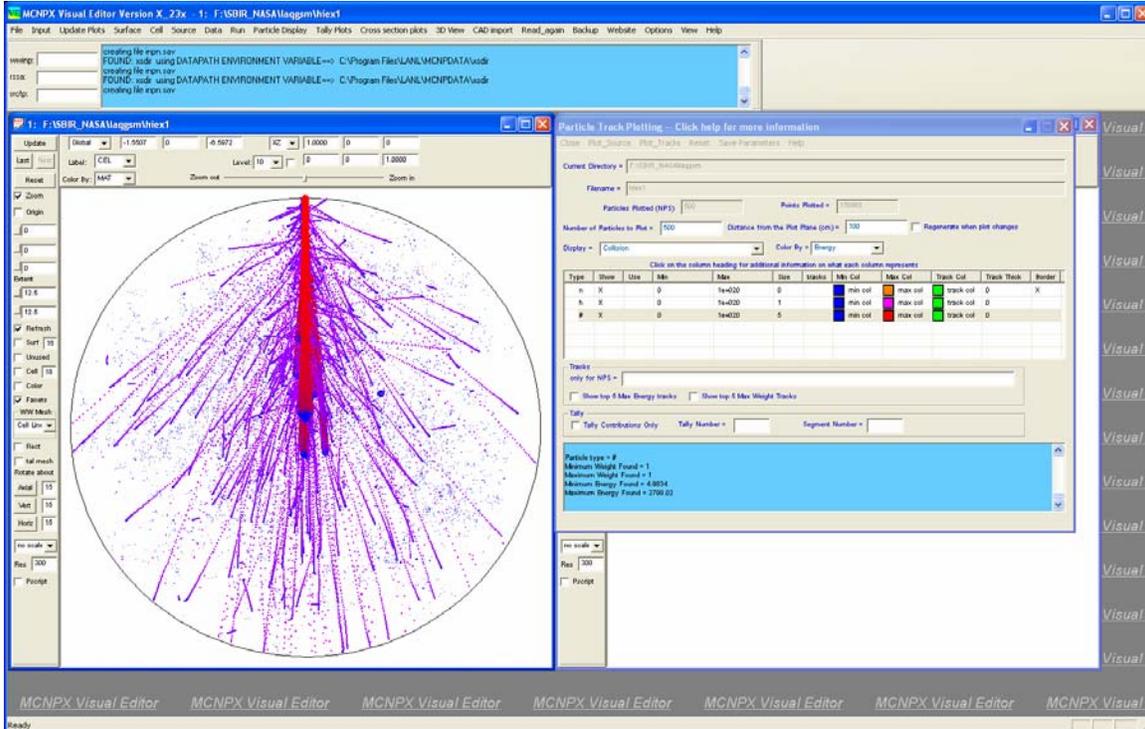


Figure 2.11. Carbon Beam Showing Secondary Protons and Neutrons.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.12 shows collision points for iron ions incident on a water phantom. The mode for this input file is shown here:

mode n h a #

The source is shown here:

```
sdef par=d3 erg=d2 sur = 3 dir=1 vec=0 0 1 pos 0 0 -1 rad=d1
si1 0 0.5
sp1 -21
si2 L 54224.00 54251.00 54278.00 $ small energy spread
sp2 0.25 0.5 0.25
si3 L 26054 26056
sp3 1 1
```

The source consist of an equal sampling of ^{54}Fe and ^{56}Fe ions

The iron ions are shown in red. The secondary protons are shown in purple. A few alphas are shown as cyan points. Secondary neutrons are shown as small green dots in the background.

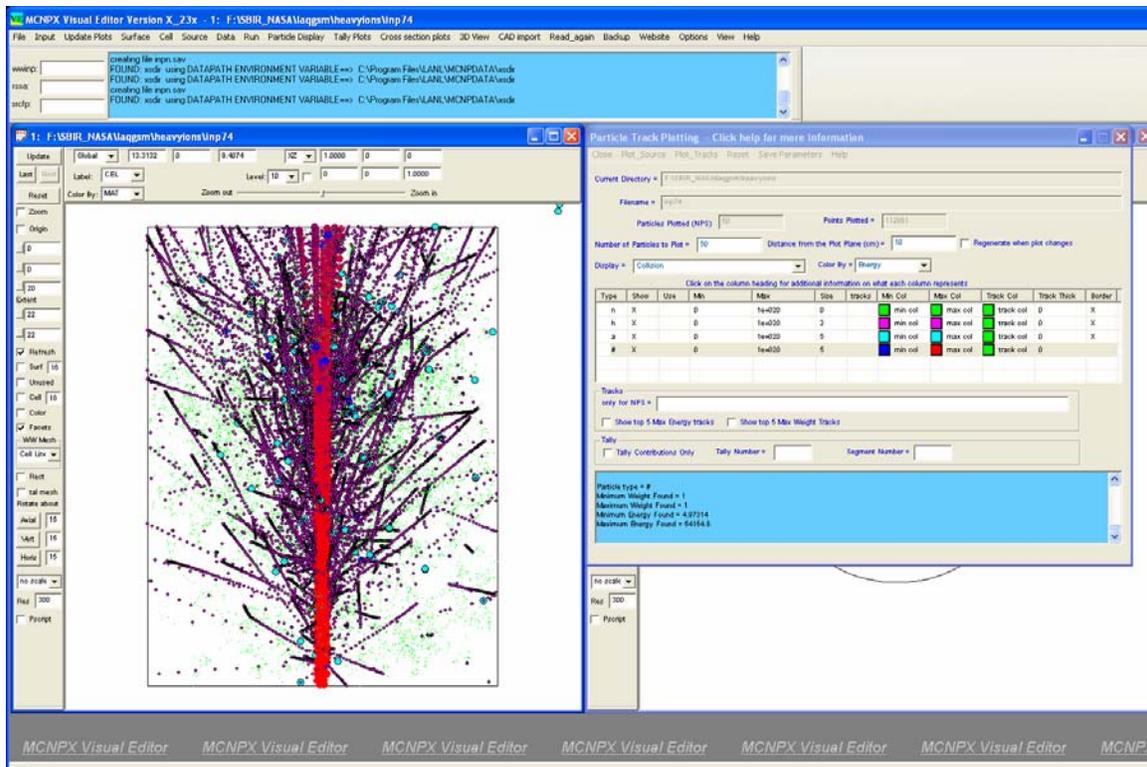


Figure 2.12. Iron Ions on a Water Phantom.

Phase 1 Final Report
 Visualization of Output Data from Particle Transport Codes

In Figure 2.13, the locations of the alpha particles have been accentuated, by setting the size of the source, protons, and secondary neutrons to their smallest size.

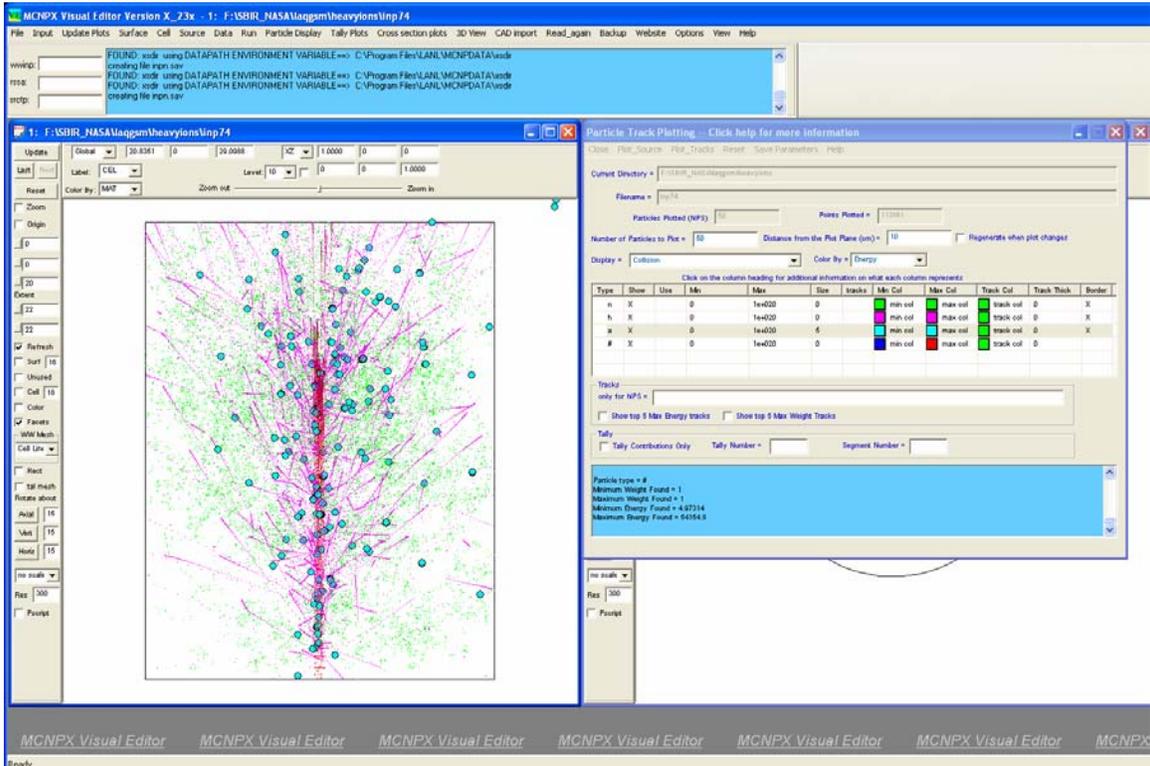


Figure 2.13. Alpha Particles Generated from Iron Ions on a Water Phantom.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.14 shows a plot of uranium isotopes in lithium. The mode for this input file is shown here:

```
mode n h d t s a #
```

The source is shown here:

```
sdef par=92238 erg=95200 pos 0 0 -2 rad=d1 axs 0 0 1 vec 0 0 1 dir 1
sil 0 0.2
spl -21 1
```

The source consist of an equal sampling of ^{238}U ions incident on a cylinder of lithium. The uranium ions come in from the bottom and lose energy as the move upward producing a smooth red to purple spectrum going from bottom to top. Secondary protons are shown in purple. A few neutrons can be seen as olive colored points. The orange collisions represent deuterons moving outward from the uranium ion beam.

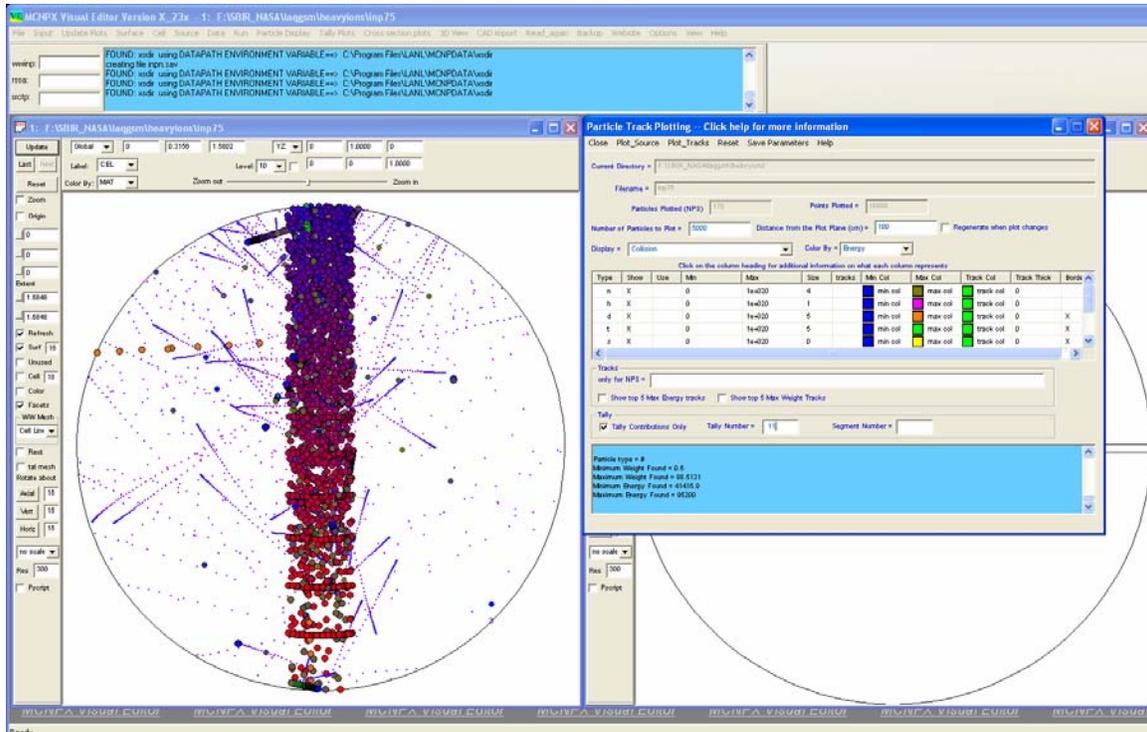


Figure 2.14. Uranium Isotopes on Lithium.

2.3 Displaying Tally Data

Figure 2.15 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 frame>” 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.

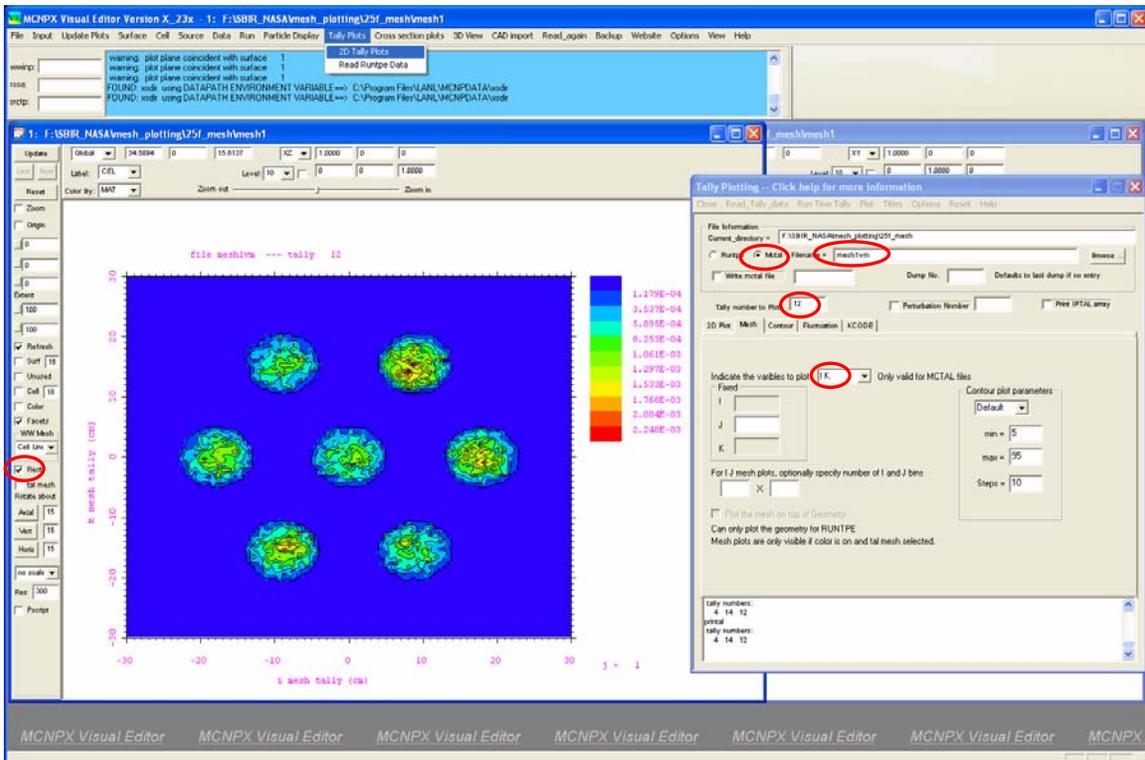


Figure 2.15. Mesh Tally Plot in the MCNPX Visual Editor.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.16 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”.

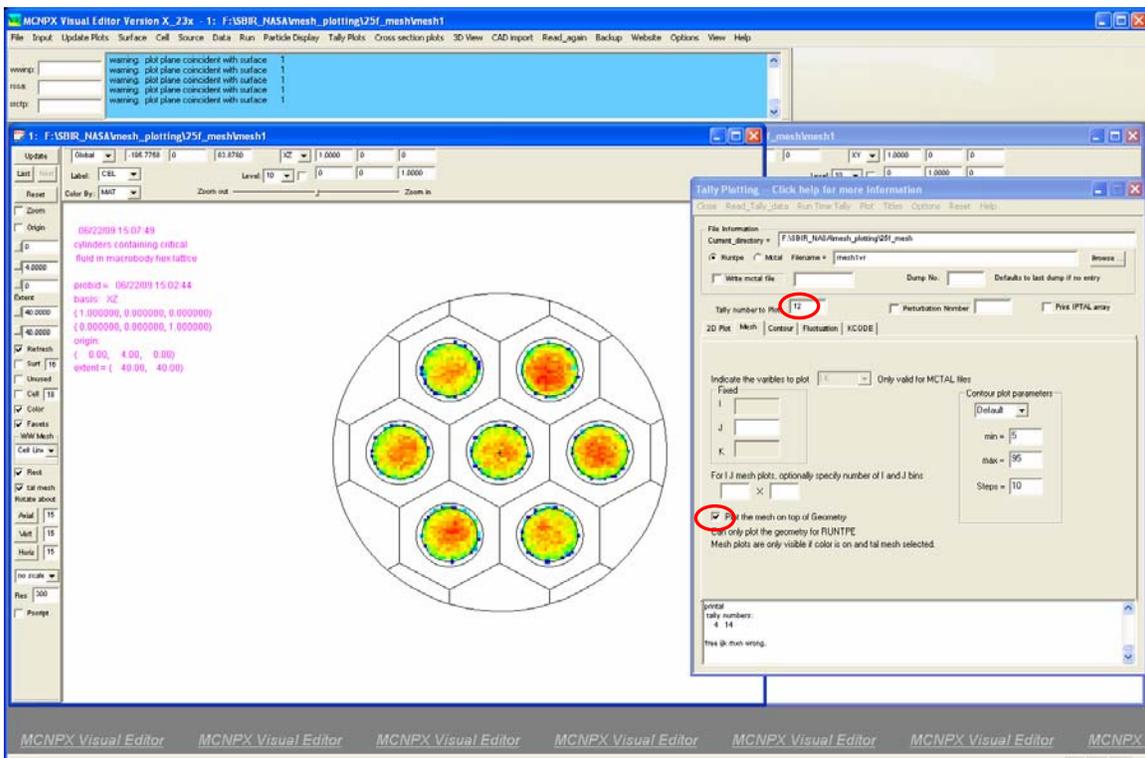


Figure 2.16. Mesh Tally Plot on Top of the Geometry.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Once the plot has been generated, the view can be changed to show different regions of the tally mesh. In Figure 2.17 the zoom toggle button was selected in the left plot and the mouse was dragged across one of the barrels to zoom in on a single barrel. The right plot shows an axial view of the tally mesh. The mesh tally grid has been turned on in both plots and is superimposed on top of the cell lines.

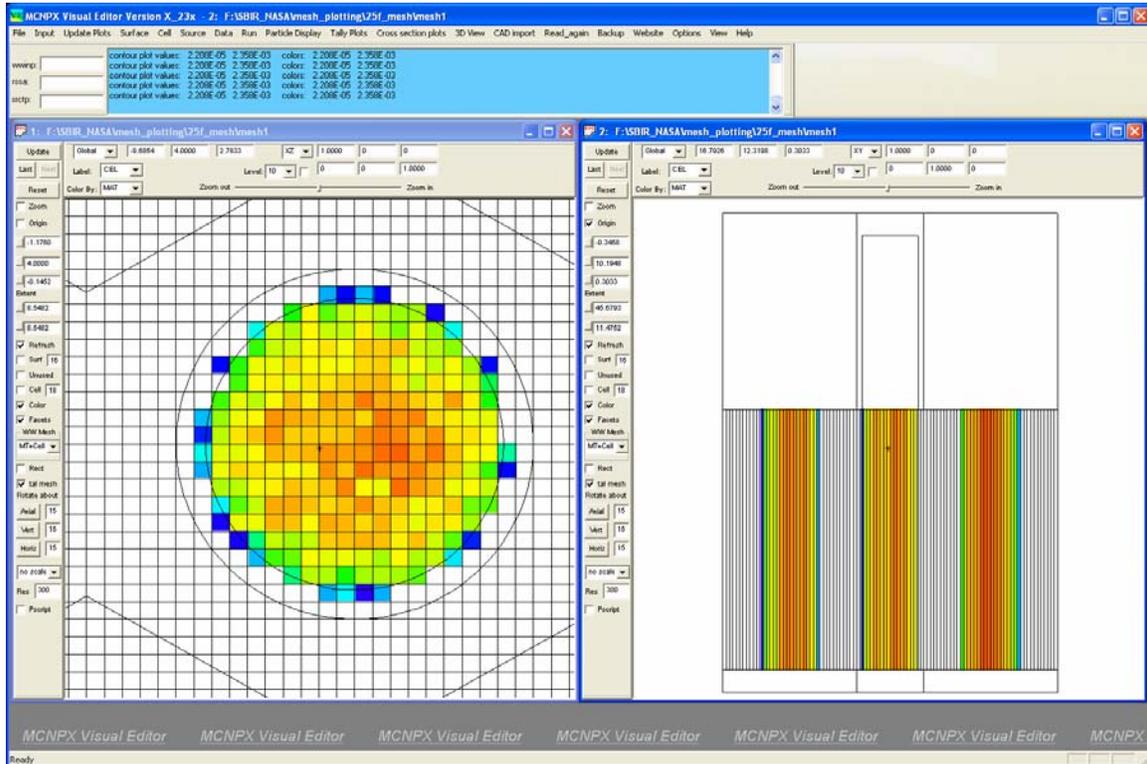


Figure 2.17. Zoomed in Region of the Mesh Tally Plot.

The latest MCNPX beta version has a number of mesh test problems contained in the testmesh directory that are 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.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.18 shows an MCNPX radiograph tally plot (tir tally) generated with the MCNPX tally plotter using the command “rnc inp01m tal 5 free sc contour line”. Figure 2.19 shows the same plot generated in the MCNPX Visual Editor.

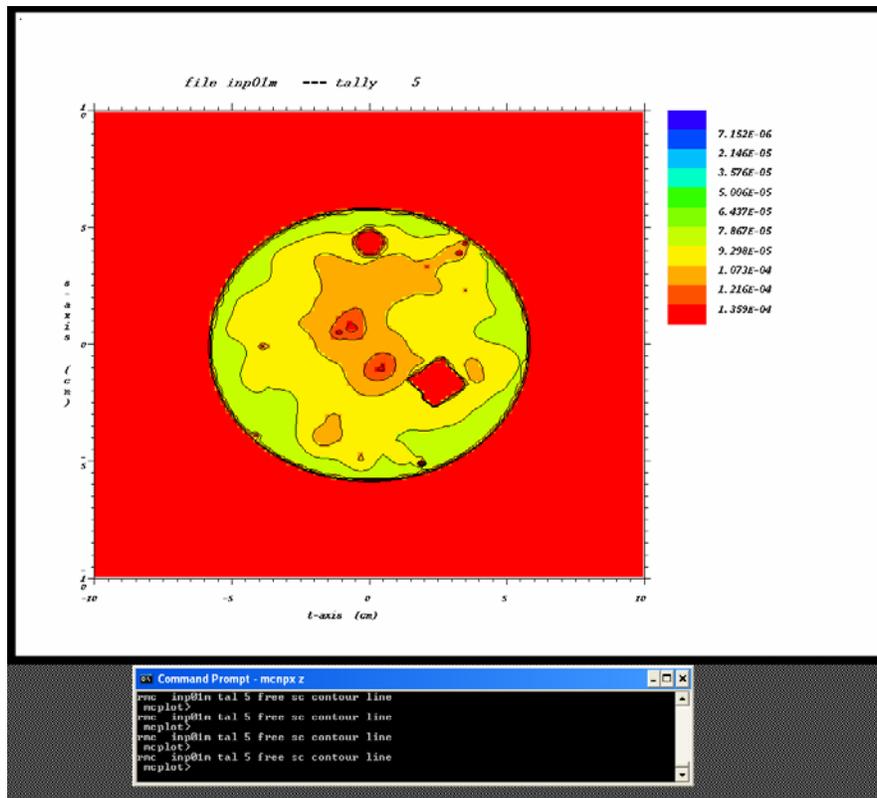


Figure 2.18. MCNPX Contour Plot.

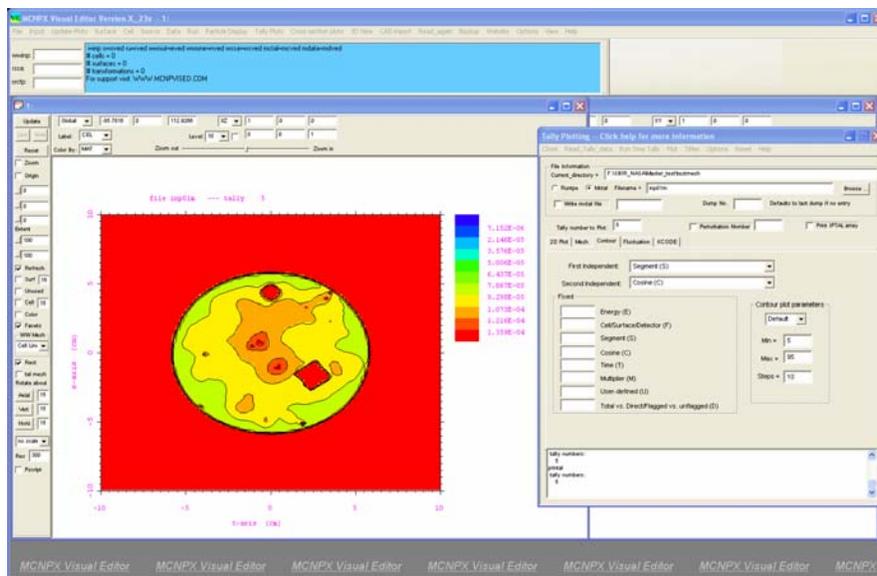


Figure 2.19. Visual Editor Contour Plot.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.20 shows a contour plot generated with the MCNPX tally plotter using the command “rnc inp01m tal 5 free sc contour line fixed d 2”. Figure 2.21 shows the same plot generated in the MNCXP Visual Editor.

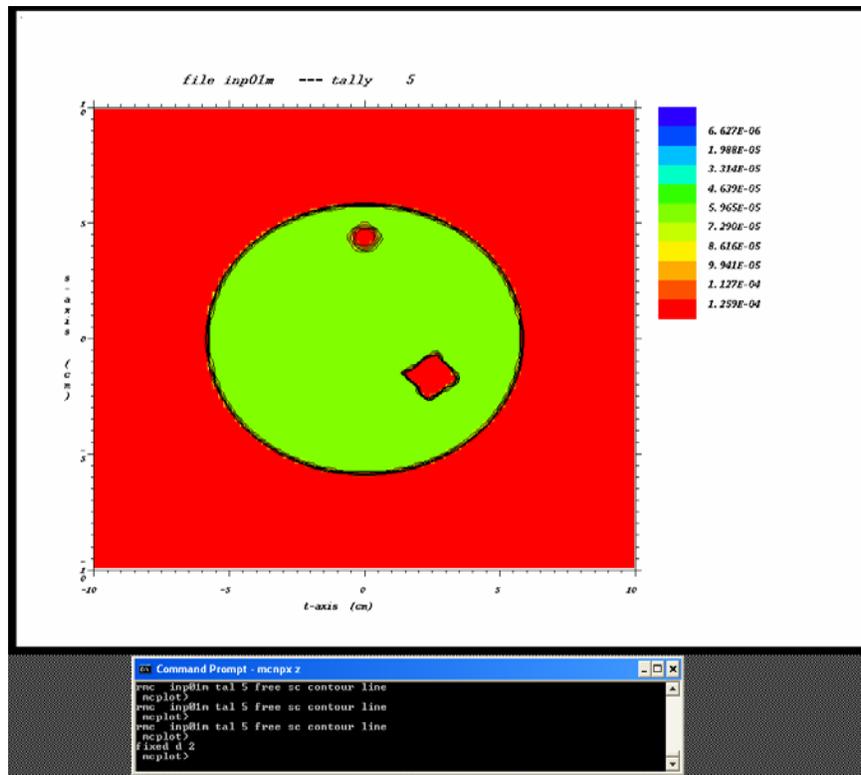


Figure 2.20. MCNPX Contour Plot with a Fixed Variable.

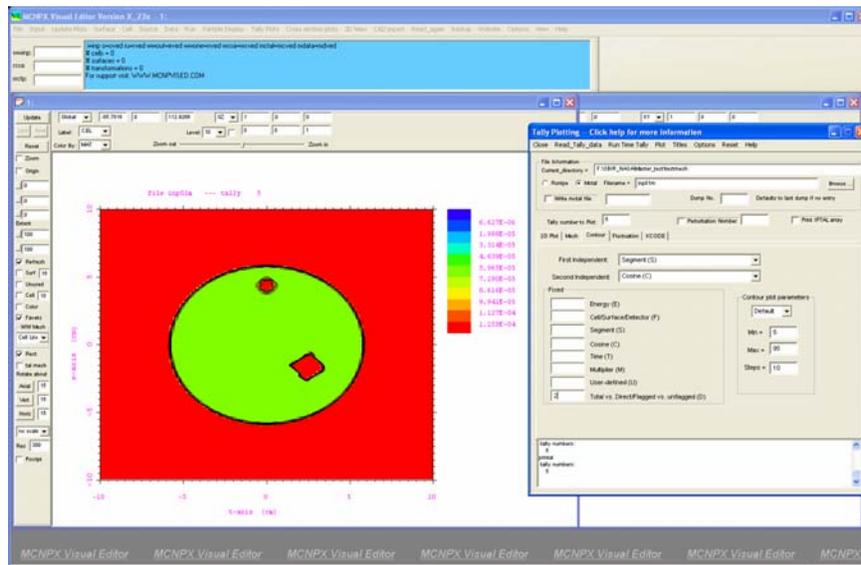


Figure 2.21. Visual Editor Contour Plot with a Fixed Variable.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.22 shows a contour plot for testmesh input file 4 generated with the MCNPX tally plotter using the command “tal 3 free ik”. Figure 2.23 shows the same plot generated in the MCNPX Visual Editor.

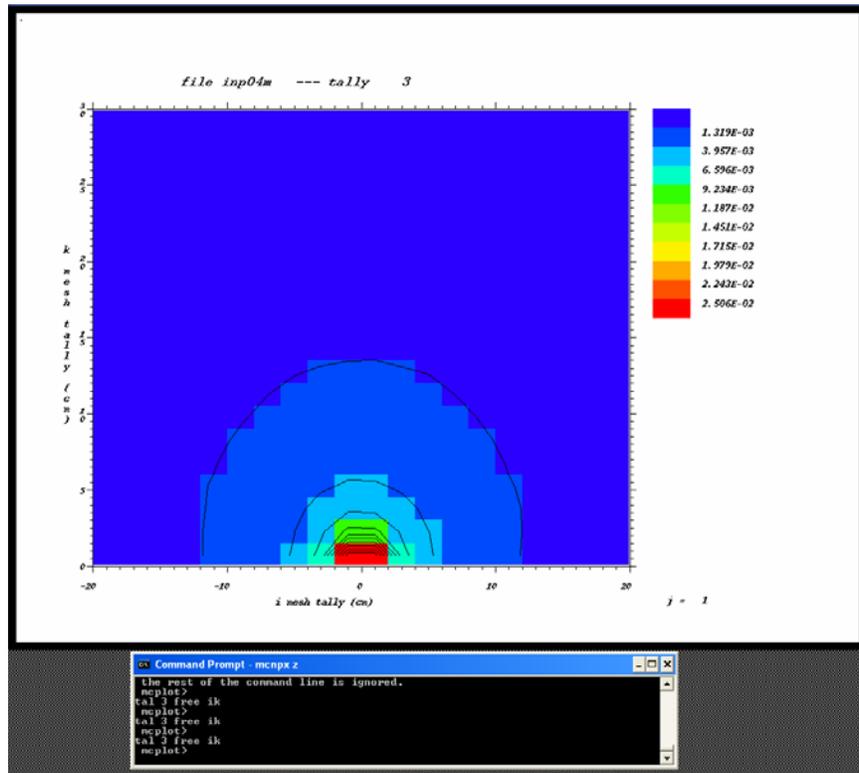


Figure 2.22. MCNPX Mesh Tally Plot.

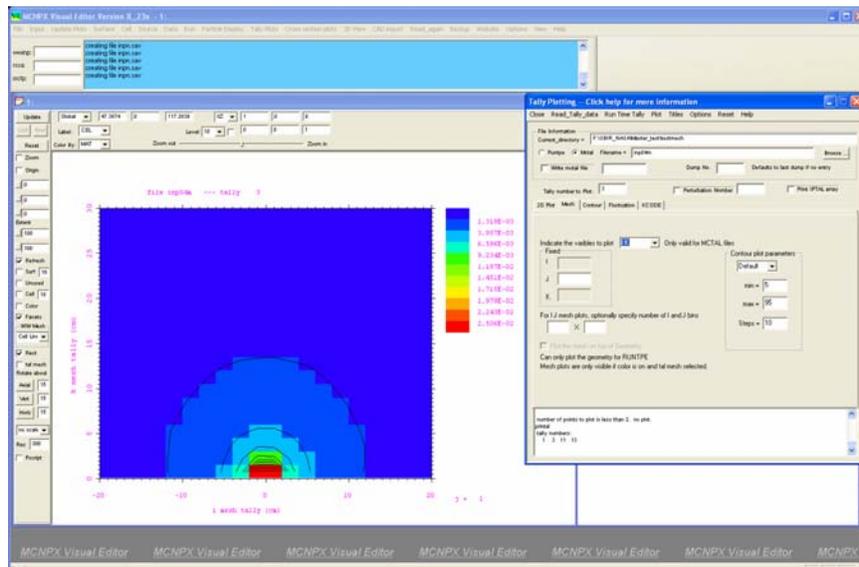


Figure 2.23. Visual Editor Mesh Tally Plot.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.28 shows a contour plot for testmesh input file 5 generated with the MCNPX tally plotter using the command “rnc inp05m tal 1 free jk contour 5 95 10 %”. Figure 2.29 shows the same plot generated in the MCNPX Visual Editor.

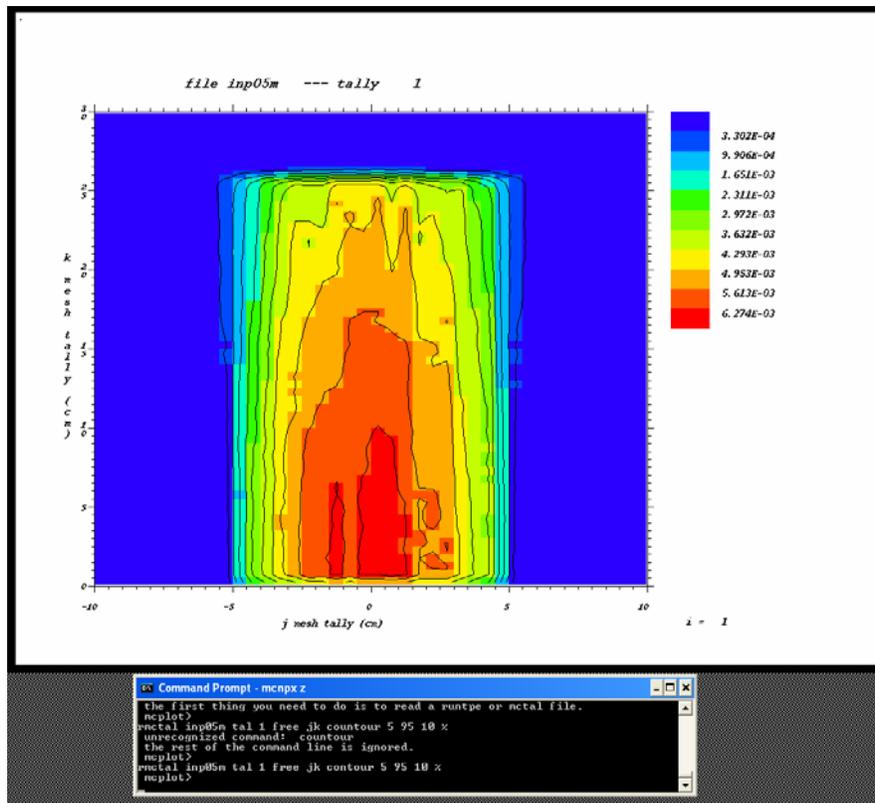


Figure 2.28. MCNPX Mesh Tally Plot.

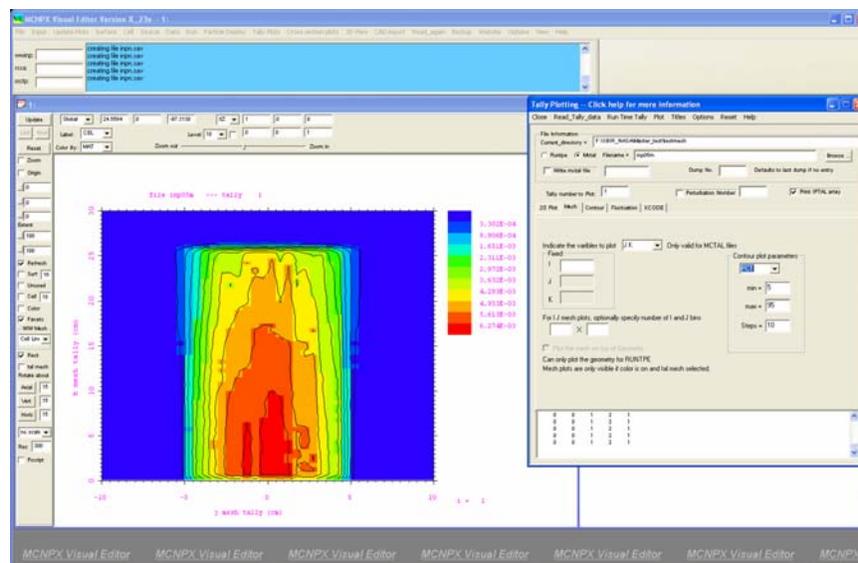


Figure 2.29. Visual Editor Mesh Tally Plot.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.30 shows a contour plot for testmesh input file 5 generated with the MCNPX tally plotter using the command “tal 13 free ij contour .01 .3 20 log”. Figure 2.31 shows the same plot generated in the MCNPX Visual Editor.

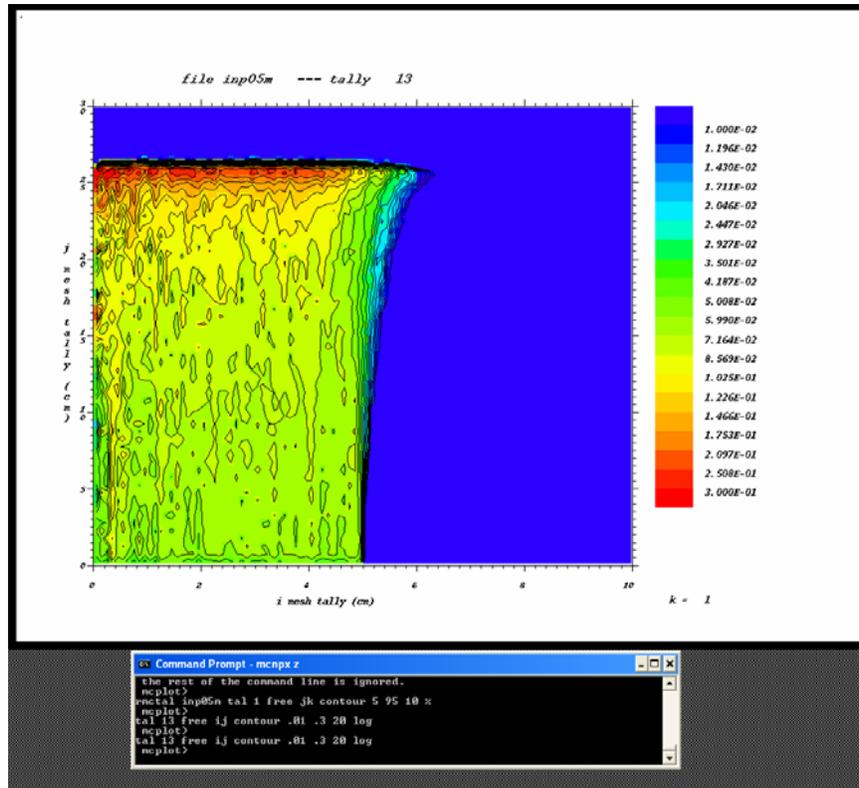


Figure 2.30. MCNPX Mesh Tally Plot.

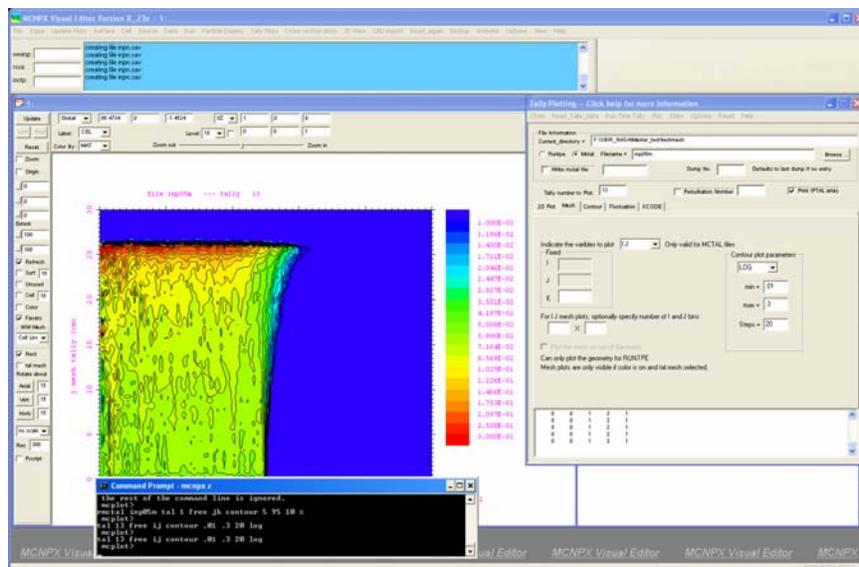


Figure 2.31. Visual Editor Mesh Tally Plot.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

Figure 2.32 shows a contour plot for testmesh input file 10 generated with the MCNPX tally plotter using the command “rnc inpl0m tal 2 free jk con 1e-5 .05 20 log”. Figure 2.33 shows the same plot generated in the MCNPX Visual Editor.

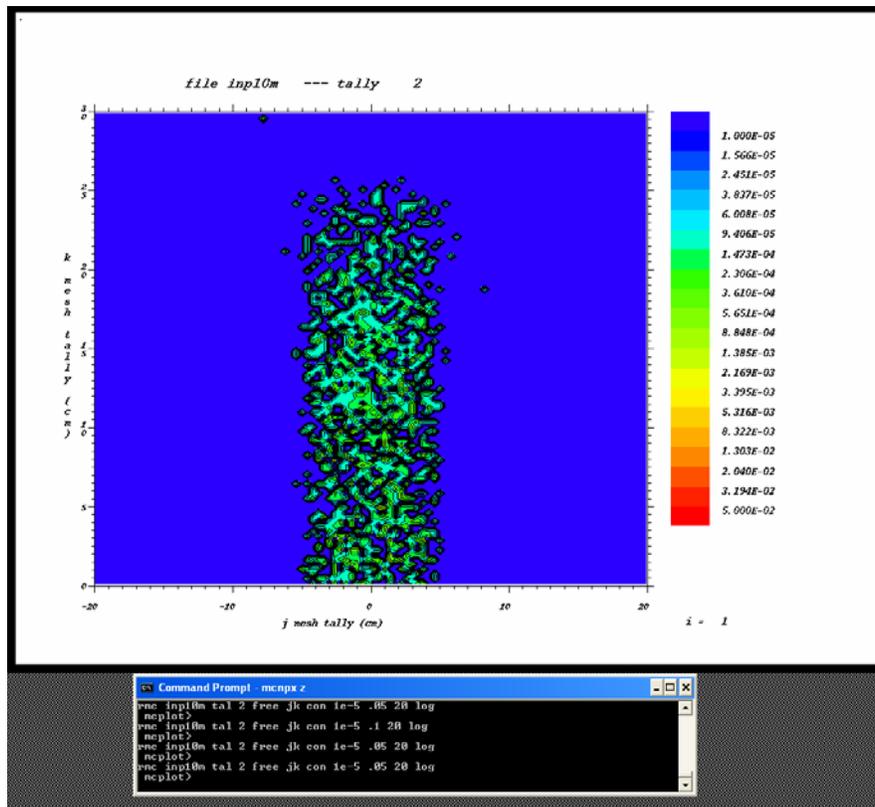


Figure 2.32. MCNPX Mesh Tally Plot.

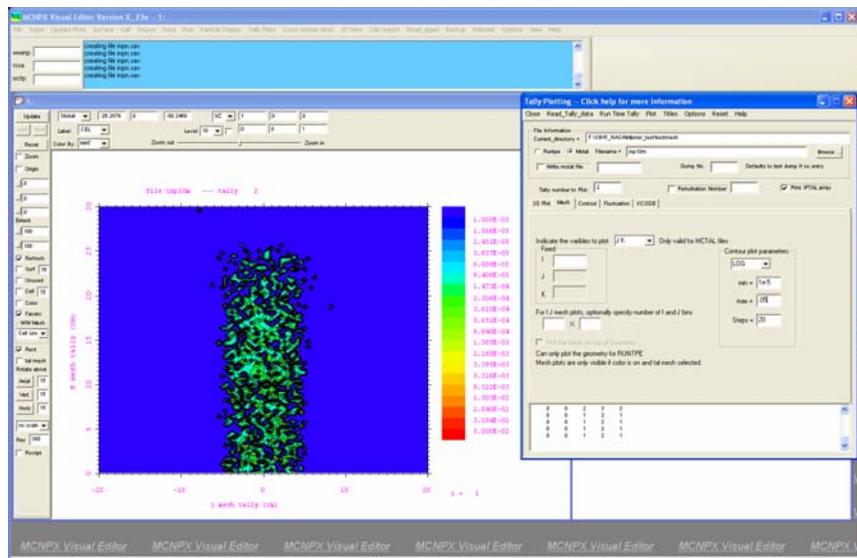


Figure 2.33. Visual Editor Mesh Tally Plot.

Phase 1 Final Report
 Visualization of Output Data from Particle Transport Codes

Figure 2.35 shows the data for table 126 that was exported from the Visual Editor that has been read into a spreadsheet. With the data in the spreadsheet, the user can now use spreadsheet formulas to manipulate the output information. This could be very useful for optimizing importances through a shield wall where the number of tracks entering each cell must be optimized.

1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	photon	activity in each cell															
2	tracks	population	collisions	collisions	number	flux	average	average	print table 126								
3	cell	entering		* weight	weighted	energy	energy	(relative)	track	weight	track	mfp					
4			(per history)	energy	energy	(cm)											
5		1	1	81	92	164	2.47E+07	0.495343	0.495343	9.65E+09	0.587119						
6		2	2	68	67	67	5.05E+06	0.311051	0.311051	9.65E+09	1.1047						
7		3	3	31	29	6	113086	0.450943	0.450943	9.65E+09	1.40906						
8		4	4	163	156	97	2.34E+08	0.378043	0.378043	9.65E+09	1.24782						
9		5	5	294	349	679	1.64E+09	0.486811	0.486811	9.65E+09	0.573835						
10		6	6	0	0	0	0	0	0	0	0						
11		7	7	1123	1074	611	5.90E+09	0.407619	0.407619	9.65E+09	5.20272						
12		8	8	2038	1905	1510	7.29E+09	0.419562	0.419562	9.65E+09	1.3057						
13		9	9	144	142	67	3.23E+08	0.430053	0.430053	9.65E+09	1.33223						
14		10	10	3	3	2	2.41E+06	0.436015	0.436015	9.65E+09	1.32746						
15		11	11	204	198	119	5.74E+08	0.42639	0.42639	9.65E+09	1.33493						
16		12	12	4	4	0	0	0.441264	0.441264	9.02E+08	0						
17		13	13	0	0	0	0	0	0	0	0						
18		14	14	2	2	1	75390.6	0.355906	0.355906	9.65E+09	1.2726						
19		15	15	38	42	63	3.80E+07	0.477198	0.477198	9.65E+09	0.55864						
20		16	16	55	48	0	0	0.377556	0.377556	6.98E+07	0						
21		17	17	111	106	118	4.45E+06	0.382288	0.382288	9.65E+09	1.2505						
22		18	18	46	46	29	273291	0.44366	0.44366	9.65E+09	1.40008						
23		19	19	0	0	0	0	0	0	0	0						
24		20	20	0	0	0	0	0	0	0	0						
25		21	21	1	1	1	37695.3	0.116127	0.116127	9.65E+09	0.486364						
26		22	22	2	1	0	0	0.120546	0.120546	3.77E+07	0						
27		23	23	0	0	0	0	0	0	0	0						
28		24	24	2	2	10	188477	0.259093	0.259093	9.65E+09	1.03171						
29		25	25	0	0	0	0	0	0	0	0						
30		26	26	4	4	0	0	0.46315	0.46315	4.74E+07	0						
31		27	27	2	2	0	0	0.438542	0.438542	1.99E+08	0						
32		29	29	114	114	0	0	0.456693	0.456693	2.90E+07	0						
33		30	31	178	210	380	1.15E+08	0.499965	0.499965	9.65E+09	0.596658						
34		31	32	373	435	820	4.95E+08	0.505631	0.505631	9.65E+09	0.604471						
35		32	33	824	972	1821	2.20E+09	0.509133	0.509133	9.65E+09	0.609372						
36		33	34	2796	3324	6247	1.51E+10	0.48703	0.48703	9.65E+09	0.573755						
37		34	38	78	89	177	2.14E+08	0.496527	0.496527	9.65E+09	0.59004						
38		35	39	40	44	92	5.55E+07	0.507219	0.507219	9.65E+09	0.605834						
39		36	40	20	24	55	1.66E+07	0.491298	0.491298	9.65E+09	0.581099						

Figure 2.35. Table 126 Data Imported into a Spreadsheet.

2.5 Advanced Display Options

Figure 2.36 shows two 3D views of a geometry with the color representing the photon importance. A cut away view is shown in the left plot and a transparent view is shown in the right plot. It can be observed that the importances are highest near the port to bias particles in this direction. The colors in the Visual Editor were set to be identical to those in MCNPX. The 3D plotting capability of the Visual Editor was then modified to use the MCNPX colors.

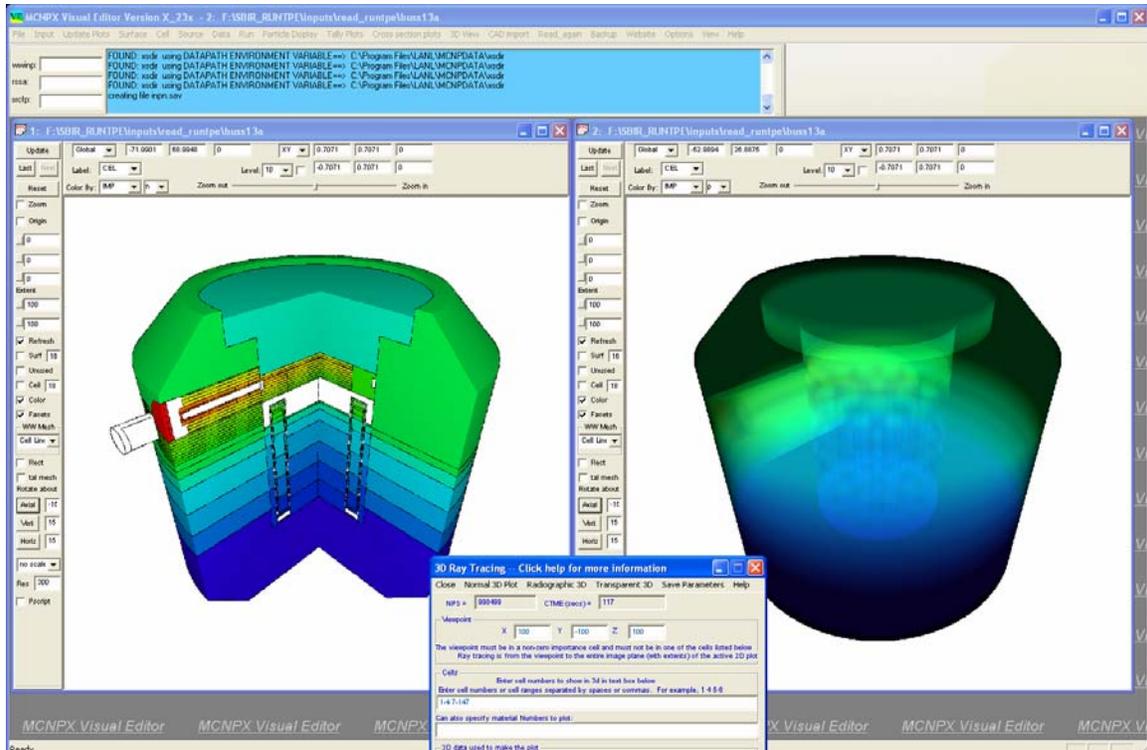


Figure 2.36. Color 3D View of Cell Importances.

3 Future Work

Future work should focus on enhancing the current capabilities. The specific future areas of research should investigate 1) exporting MCNP output data; 2) advanced tally options; and 3) advanced display options.

3.1 Exporting MCNP Output Data

Phase 1 demonstrated the feasibility and usefulness of exporting MCNP output data to an external file that can be read by a spreadsheet program. This capability should be investigated further to adapt it to other data tables that may be of use. In particular the export of tally data so that it can be incorporated into an external program should be investigated and implemented. The primary goal would be to help the user understand and extract important information from the MCNP run.

3.2 Advanced Tally Options

Monte Carlo solution techniques have the advantage of sampling a wide distribution of tally contributions, in addition to providing mean values and associated relative errors. In most applications, tally contributions come from two competing components, one that contributes to the desired signal and one that distorts the signal (noise). Recent tally tagging capabilities within MCNP allow these signal and noise components to be separated. It is now desirable to be able to view the distribution of tally contributions, often called probability distribution functions (PDFs), independently for signal and noise contributions. Furthermore, a capability is needed that can manipulate these PDFs in such a way to produce graphical forms that characterize a detection system's effectiveness. One approach to this is called receiver-operator characterization (ROC), where probability of false (i.e., noise) detection is plotted against probability of detection. ROC curves enable one to evaluate and optimize a wide variety of detection systems.

Future work should investigate adding a post-analysis graphics capability that can read and manipulate tally and PDF data generated by MCNPX. This capability will include the production of ROC curves that are formed from the integration of multiple PDFs. Appropriate ROC-curve error estimates will be computed and displayed, using the fundamental Monte Carlo statistical information.

3.3 Advanced Display Options

Advanced display features should be investigated and implemented include generating 3D plots showing particle activity. The display of particle tracks in 3D and the 3D color display of mesh tallies.

Phase 1 Final Report
Visualization of Output Data from Particle Transport Codes

The Visual Editor can already display 2D particle track data. The dynamic 3D view that is currently available in the Visual Editor should be modified to show the particle collision points in 3D. The 3D location of each collision point is known. This information can be passed to the graphical user interface to be displayed. An example of this kind of plot is shown in Figure 3.1. In this Figure, the collision points are shown as black dots superimposed over a 3D image of the geometry. This capability can be very useful in understanding the physics of the problem being analyzed.

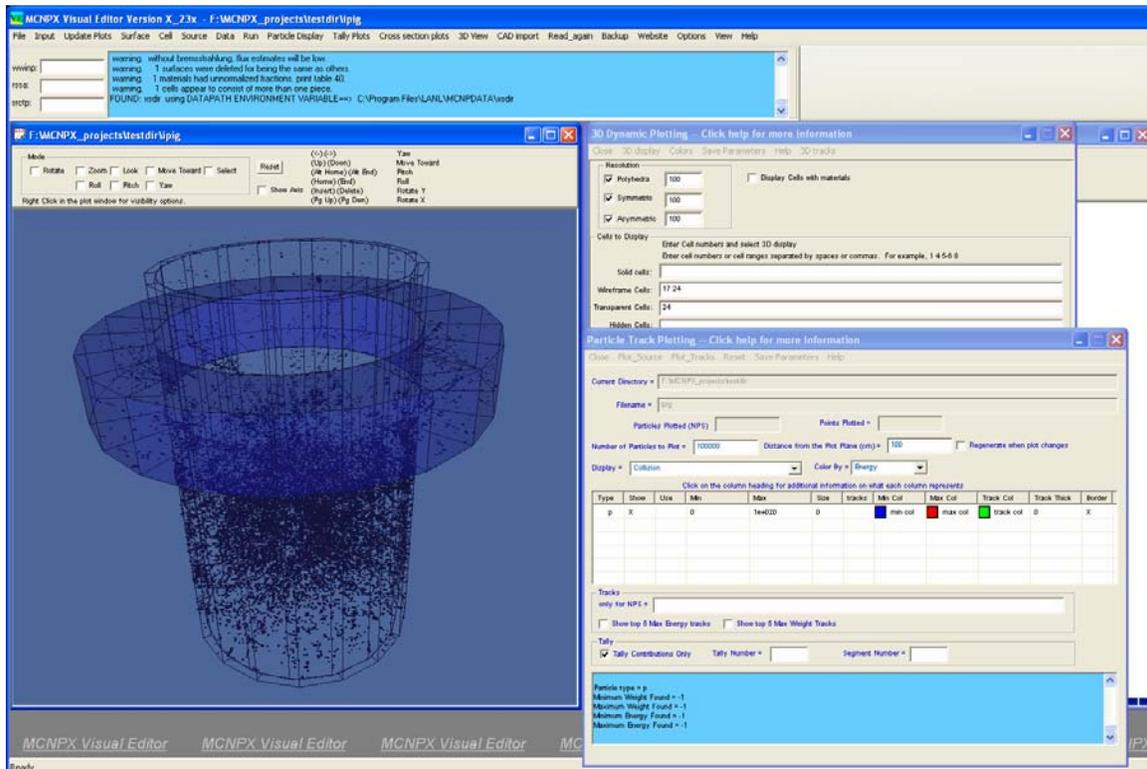


Figure 3.1. 3D Display of Collision Points.

An investigation should also be made into the best way to display mesh tally data on the 3D surfaces of the geometry and/or generate a 3D array of voxels, with each voxel representing the color for the data for that region as calculated by MCNP. These voxels could then be made semi-transparent to show a 3D display of the mesh data.